Umwandlung des Alpha-Registers in Speicherregister 41C

Jedem 41C Besitzer ist das Alpha-Register bekannt. Es trägt mit dazu bei, daß der hp-41c als momentan leistungsfähigster Taschenrechner auf dem Weltmarkt angesehen werden darf. Die Vorteile des Alpha-Registers sind jedem Anwender mehr oder weniger bekannt.
Was aber ist mit Programmen oder kurzen Routinen, die dieses Register nicht benutzen? In diesem Fall läßt sich das Alpha-Register völlig gefahrlos in 3 Speicherregister verwandeln. Diese Register heißen STO M , STO N , STO O .
Die Verwendung dieser Register möchte ich an einem von mir entworfenen Programm zur Berechnung von Größtem gemeinsamen Teiler und Kleinstem gemeinsamen Vielfachen (GGT & KGV) erläutern.
Zur KGV Berechnung ist ein Register zum Abspeichern eines Zwischenwertes erforderlich. Wird diese Routine aber von einem anderen Programm aufgerufen, so wird der Inhalt des verwendeten Registers zerstört und das Hauptprogramm wird fehlerhaft fortgesetzt.
Dieses läßt sich umgehen, indem zur Zwischenspeicherung das Register M verwandet wird, da keine Alpha-Operation während der KGV Berechnung durchgeführt wird.
Das GGT & KGV Programm lautet sodann:

```
Lbl GGT
MOD
LAST X
X<>Y
X≠0?
GTO GGT
RDN
RTN
Lbl KGV
STx Y
X<>Y
STO M
X<>Y
ST/ Y
XEQ GGT
RCL M
X<>Y
/
END
```

So weit, so schön. Aber wie lassen sich nun diese Befehle erzeugen?
Sofern die Gerätenummer Ihres Rechners über 195o liegt - Gar nicht. Sie können sich dann die Befehle nur von mir auf Magnetkarte schicken lassen und dann einlesen, was allerdings auch wesentlich bequemer ist.
Sollte Ihre Gerätenummer kleiner gleich 195o sein, so geben Sie folgendes Programm ein, nachdem Sie Master Clear durchgeführt haben:

```
01 +
02 +
03 +
04 +
:     (2oo mal "+")
:
:
200 +
201 +
202 999
203 STO 00
204 1,917585
205 STO IND 00
```

Positionieren Sie den Rechner mittels GTO.000 und drücken Sie R/S. Nach einigen sec. Rechenzeit können Sie nach Zeile 170 gehen und werden dort den Befehl STO M vorfinden.
Sie können auch die folgenden Befehle erzeugen, indem Sie genau wie oben vorgehen, jedoch in Zeile 204 schreiben: 1,907585 für RCL M 1,917685 für STO N 1,907685 für RCL N 1,917785 für STO O 1,907785 für RCL O .

Dies ist zugegebener Maßen eine äußerst umständliche Methode. Wünschen Sie daher, alle Befehle auf Magnetkarte zu erhalten, so schicken Sie mir bitte (egal welche Gerätenummer Ihr Rechner hat) eine Magnetkarte und überweisen die übliche Pauschale von 1,5o DM auf das Ihnen bekannte momentane Konto des Clubs. Sie werden dann zusammen mit den nächsten Clubinformationen eine Kopie der obenstehenden Befehle zuzügl. X<>M X<>N X<>O erhalten. Weitere Tricks dieser Art in den nächsten Clubinfos.

OLIVER RIETSCHEL , EUTIN

Referenz: William C. Wickes ; Bericht in PPC Journal Oct. 1979 , Seite 32

Fortsetzung: Umwandlung des Alpha-Registers in Speisherregister ..41c

Im Prisma 2 wurde die Erzeugung der Befehle STO M,N,O und RCL M,N,O beschrieben. Es lassen sich weitere Befehle auf diese Art und Weise erzeugen, indem in Zeile 2o4 des aufgeführten Programmes folgende Codes geschrieben werden:

Allgemeine Form: 1,9XYZ85

X gibt hierbei die Art des Befehles an
YZ gibt den Code für das Register an.

X =	ergibt		YZ =	ergibt Register	
o		RCL	66		A
1		STO	67		B
2		ST+	68		C
3		ST-	69		D
4		STx	! 7o		T
5		ST/	71		Z
6		ISG	72		Y
7		DSE	73		X
8		VIEW	74		L
9		ΣREG	75		M
			76		N
			77		O
			78		P
			79		Q

V O R S I C H T !! Nicht alle Befehle lassen sich uneingeschränkt verwenden. Es ist möglich, daß z.B. Flags, Programme, Tastenbelegungen u.s.w. verändert werden.

Bisher habe ich folgendes festgestellt:

M,N,O,P sind Alpha-Register (Vorsicht bei P)

X,Y,Z,T,L sind Stacks und Last X

Dieses Thema ist hiermit sicherlich noch längst nicht abgeschlossen, doch habe ich momentan nichts wesentliches mehr auf Lager. Immer noch stecken Geheimnisse in unseren Einundvierzigern. Deshalb bitte ich Sie: Schreiben auch Sie Ihre Erfahrungen und Entdeckungen!!!! Mich beschäftigt folgendes Problem sehr: Wie kann man diese Befehle irgendwelchen Tasten zuordnen??? !?

OLIVER RIETSCHEL , EUTIN
GERT-JÜRGEN GRÜNLER , BERLIN

PS: Die Befehle zu den Registern M,N,O sind bereits auf Karte für alle Rechner erhältlich. Außerdem noch die Austauschbefehle X gegen M,N oder O !!

Programme 41c

Mondphasen/ Finsternisse ; Annuitätentilgungsplan ; Telefongebührenzähler ; Timer und Uhr

1)
Phasen/ Finsternisse berechnet nach Eingabe von JJJJ,jj durch "A" die nächste Finsterniss, durch "B" die nächste Phase.
"C" = Wiederholung der Anzeige mit Datum, Uhrzeit, Bedeckungsgrad und Angabe des Ereignisses.
"E" = nächstes Ereigniss (von A oder B)

Dieses Programm ist ein 67 - Programm von H. Schnepf und Lunnebach, das ich für den 41c umschrieb und mit Text versah.

2)
Tilgungsplan: Achtung: Nach Prompting (Bedeutung wie Standardsammlung) nicht R/S drücken! Aber: Wenn, zügig die geforderten Werte eingeben!

Restschuld,. PMT, Summe der Zinsen und Anteil der Zinsen von PV in %, Effektivverzinsung bei Damnum.

3)
Telefongebühren: RTN , R/S , Sekunden/Einheit eintippen (zügig!) und warten bis "/ R/S" . Dieses (R/S) dann tun, wenn der Teilnehmer sich meldet.

4)
Uhr: Eingabe (vor erreichter Zeit): HH,MMSS ; dann "A" und für endgültigen Start schließlich: R/S

5)
Timer: Piepst nach Ablauf der eingestellten Zeit (wie oben) u. "E" , R/S

Sie finden die Programmlistings auf den beiden folgenden Seiten !

GÜNTER LELARGE

Ein Vorschlag:
Um umständliche Bestellung und Verwaltungsaufwand von Programmen zu vermeiden, könnten diese gleich mit den Infos abgedruckt werden. Dabei sollte es dem Ersteller überlassen bleiben, ob er bloß eine Bedienungsanleitung beifügt, oder eine "Seminararbeit" abliefert, die die Zeitersparnis durch das Programm zunichte macht.

GÜNTER LELARGE

Hierzu:
Prinzipiell stimme ich diesem Vorschlag zu, doch frage ich mich, welchen Reiz es für Mitglieder haben soll, wenn Programme nicht vergütet werden (können) . Wie ist Ihre Meinung hierzu ?

OLIVER RIETSCHEL

Wieviel Mitglieder ?

Mich würde interessieren, wieviel Mitglieder unser Club bisher hat und wieviele aus dem Ausland kommen ! EDMUND WEITZ

Am 11/o5 besaß der Club genau 25 Mitglieder. Einen Österreicher, einen Italiener und einen Franzosen, der wegen 'Kommunikationsschwierigkeiten' wieder ausgetreten ist. O.R.



PHASEN/
FINSTERNISSE

```
     STARTDATUM: JJJJ.
JAHRE (DEZIMAL)
A: FINSTERNISSE
B: MONDPHASEN
C: WIEDERHOL. D. ANZEIGE
E: NAECHSTES EREIGNIS
                 PRP "

 01•LBL 14
 02•LBL "WEITER"
 03•LBL E
FS? 00  "PHASEN"  FC? 00
"FINSTERNISSE:"  FC? 55
AVIEW  RCL 01  RCL 02
SIN  *  ST+ 00

 15•LBL 03
9.1  STO 25  XEQ 00
ENTER↑  ISG 25
RCL IND 25  *  .2  P-R
ISG 25  RCL IND 25  *
X<>Y  STO 04  *  +
STO 05  XEQ 00  STO 03
XEQ 00  STO 02  -  +  +
SIN  32  %  +  RCL 04  +
ST+ 03  RCL 03  RCL 04
RCL 24  D-R  /  +
ST+ 02  RCL 02  .7
ST* 04  P-R  9  +  *
ST+ 03  RCL 03  RCL 05
+  3  LN  P-R  2  D-R  -
*  ST+ 03  RCL 03
RCL 04  XEQ 00  -  +
STO 04  ENTER↑  +  SIN
9  /  -  ST+ 05  RCL 06
RCL 05  TAN  ATAN  ABS
X<=Y?  GTO 12  LASTX
RCL 02  COS  8  -  *  1
%  ST+ 00  GTO 03

 103•LBL 00
ISG 25  RCL IND 25
RCL 00  *  ISG 25
RCL IND 25  +  RTN

112•LBL "PHASEN"
FC? 04  XEQ J  SF 00
CF 03  "PHAS: "  GTO 04

119•LBL "FINSTE"
FC? 04  XEQ J  "FINST: "
SF 03  CF 00

125•LBL 04
SF 26  CF 29  ARCL X
FC? 55  AVIEW  RCL 23  *
RCL 21  -  STO 00
GTO 03

 137•LBL 12
 138•LBL "WJM"
 139•LBL C
 CF 02  RCL 09  RCL 04
 SIN  *  RCL 05  ENTER↑
 +  RCL 04  -  SIN  7  /
 +  ABS  RCL 05  COS
 X<0?  SF 02  FS? 00
 GTO 01  RCL 08  *  -
 RCL 02  COS  RCL 06  *
 RCL 07  +  -  X>0?
 GTO 14  LASTX  RCL 21
 FRC  *  /  BEEP  FS?C 02
 GTO 09  "SONNF:"  GTO 13

 183•LBL 09
 CLA  "MONDF: "

 186•LBL 13
 FIX 2  ARCL X  FC? 55
 AVIEW  PSE  PSE

 193•LBL 01
 RCL 00  INT  ENTER↑
 ENTER↑  RCL 22  -
 RCL 23  /  INT  STO 25
 RCL 23  *  INT  -
 ENTER↑  ENTER↑  RCL 24
 /  INT  STO 20  RCL 24
 *  INT  -  12  RCL 20  1
 -  X<=Y?  GTO 02  X<>Y
 -  ISG 25

 227•LBL 02
 1  %  R↑  +  RCL 25
 1 E6  /  +  RCL 22  FRC
 +  FS?C 02  GTO 10
 FS? 03  GTO 11  " NEUM:"
 GTO 11

 245•LBL 10
 " VOLLM:"

 247•LBL 11
 BEEP  "├ AM "  FIX 6
 ARCL X  AVIEW  PSE  PSE

 255•LBL 12
 CLA  RCL 00  FRC  24  *
 HMS  FIX 2  FS? 01  RTN
 "├ UM "  ARCL X  "├ UHR"
 AVIEW  CF 01  FC? 55
 RTN  GTO 14

 274•LBL "START"
 SF 04  STO L  CLRG
 14.765  STO 01  .07
 STO 06  1.23  STO 07
 .27  STO 08  5.14528
 STO 09  693986.43
 STO 21  122.0019  STO 22
 365.25  STO 23  30.6001
 STO 24  -.985647  STO 10
 142.9  STO 11  282.2
 STO 12  9.6  STO 13
 13.176398  STO 14
 153.17  STO 15  -.111487
 STO 16  32.83  STO 17
 -.852952  STO 18  262.49
 STO 19  RCL L
 "  ReadY"  TONE 5
 FC? 55  AVIEW  .END.
```

ANNUITAETEN-
TILGUNGSPLAN

MIT DAMNUM

```
 01•LBL "TILG"
CF 28  "FV?"  XEQ 14
STO 01  "%?"  XEQ 14
STO 02  "N?"  XEQ 14
STO 03  "AUSZ. IN %?"
XEQ 14  STO 04  RCL 01
XEQ 01  /  RCL 11  *
STO 15  "PMT="  ARCL X
PROMPT  STO 10  RCL 15
-  RCL 04  1 E2  /
RCL 01  *  "%eFF="
AVIEW  STO 09  0  STO 15
```

ANNUITAETEN-TILGUNGSPLAN
FORTSETZUNG

```
 37•LBL 08
RCL 15  ST+ 00  XEQ 09
RCL 10  RCL 12  RCL 11
*  RCL 13  /  RCL 09  *
-  RCL 11  RCL 14  *
RCL 03  *  RCL 12  +
RCL 13  *  RCL 11
RCL 12  *  RCL 14  *
RCL 03  *  -  RCL 13
X↑2  /  RCL 09  *  /
STO 15  ABS  1 E-5
X<=Y?  GTO 08  RCL 15
ST+ 00  RCL 08  1  -
1 E2  *  ARCL X  PROMPT
GTO 08

 83•LBL 09
RCL 08  STO 11  RCL 03
Y↑X  STO 12  STO 13
RCL 00  /  STO 14  1
ST- 11  ST- 13  RTN

102•LBL 01
RCL 02  1 E2  /  1  +
STO 00  RCL 03  Y↑X
STO 11  1  -  RCL 00  1
-  /  RTN

119•LBL 08
RCL 03  STO 13  0
STO 14

124•LBL 06
1  ST+ 14  "NACH "
FIX 0  RCL 14  ARCL 14
"⊢ JAHR:"  AVIEW  PSE
FIX 2  STO 14  STO 03
XEQ 01  RCL 10  *  CHS
RCL 11  RCL 01  *  +
CLA  ARCL X  "⊢ DM"
AVIEW  PSE  PSE  PSE
PSE  X>0?  GTO 06
RCL 10  RCL 14  *  +
"Σ="  ARCL X  AVIEW  PSE
PSE  RCL 01  X<>Y  ΣCH
RCL 13  STO 03  RDN
"ΣV.PV="  ARCL X  AVIEW
STOP

174•LBL 14
AVIEW

176•LBL 15
PSE  FS?C 22  RTN
GTO 15  GTO 14  .END.
```

TELEFONGE-BUEHRENZAEH-LER

PROMPT OHNE R/S • END-GUELTIGER START DANN MIT

R/S
PRP **

```
 01•LBL "TEL"
"SEK/EINHEIT?"  AVIEW

 04•LBL 11
PSE  FC?C 22  GTO 11
8.7  *  1 E3  /  FIX 3
RND  FIX 2  STO 02  CLX
CLA  STO 01  "↑ R/S"
TONE 9  PROMPT  GTO 01

 23•LBL 05
X<>Y  ISG 00  GTO 05

 27•LBL 01
.23  ST+ 01  CLA  "  "
ARCL 01  "⊢ DM"  TONE 4
AVIEW  RCL 02  STO 00
GTO 05  END
```

TIMER: E

EINGABE: HH,MMSS, R/S

UHR: A

```
 01•LBL A
CLRG  46  STO 21  FIX 4
RTN  RDN

 08•LBL 05
VIEW X  HR  720  1/X  +
STO 20  RCL 21  RCL 25
+  INT  STO 25

 20•LBL 01
DSE 25  GTO 01  RCL 20
HMS  GTO 05

 26•LBL E
STO 04  INT  60  *  262
*  RCL 04  FRC  STO 11
1 E2  *  INT  262  *  +
RCL 11  1 E2  *  FRC
1 E2  *  60  /  262  *
+  INT  STO 25  STOP
FIX 0

 57•LBL 03
VIEW 25  DSE 25  GTO 03
10  STO 10

 63•LBL 04
TONE 9  TONE 8  DSE 10
GTO 04  .END.
```



Bei meinen Programmen, die alle auf Drucker ausgelegt sind, kommt es zu immer wiederkehrenden Programmteilen:

1) LBL FT — Formatieren der Ausgabe
2) LBL PS — Test, ob Drucker vorhanden
3) LBL SZ — Test, ob genuegend Datenregister vorhanden (wird nach PS automatisch mit durchgeführt)
4) LBL -- — unterstreichen einer ganzen Zeile

Der Druckertest ist nicht unbedingt notwendig, da beim ersten Druckerbefehl (ausgenommen ADV) sowieso NONEXISTENT auftaucht. Der Speicertest ist eher wichtig, da SIZE nnn innerhalb eines Unterprogrammes ausgeführt, die Rücksprungadressen löscht. Formatieren ist für den Druck von Tabellen schön.

```
                 PRP "FT"

 01+LBL "FT"
X<>Y  STO Z  ABS  RND
X<>Y

 07+LBL 00
10↑X  X<=Y?  GTO 01
ACA  RDN  LASTX  DSE X
GTO 00

 17+LBL 01
RCL Z  RND  ACX  LASTX
RTN

 23+LBL "PS"
FS? 55  GTO "SZ"
"PRINTER"  SF 11  PROMPT
CF 11

 30+LBL "SZ"
SF 25  RCL IND X  RDN
FS? 25  RTN  R↑  FIX 0
1 +  "SIZE "  10  X>Y?
"⊢0"  X↑2  X>Y?  "⊢0"
RDN  ARCL X  SF 27  RTN

 51+LBL "--"
SF 12  "-------"  ASTO L
ARCL L  PRA  CF 12  END

                 CAT 1
LBL'FT
LBL'PS
LBL'SZ
LBL'--
END          128 BYTES
.END.        05 BYTES
```

Wirkungsweise und Programmablauf

<u>FT:</u> Der zu druckende Wert steht im Y-Register, im X-Reg ist die Anzahl der vor dem Dezimalkomma (oder -punkt) -2 bereitzuhaltenden Druckpositionen, z. B.: die Zahl 4562.65 soll mit 7 Stellen vor dem Punkt geschrieben werden: 4562.65 ENTER 5 XEQ "FT" PRBUF
FT wird jetzt 5-mal durchlaufen, oder solange, bis 10^n kleiner-gleich der zu druckenden Zahl ist. Bei jedem Durchlauf wird eine Leerstelle in das Druckpuffer gegeben. Die Zahl wird auf die eingestellte Nachkommastelle gerundet ausgegeben, die 'Originalzahl' steht in LAST X. Positive und negative Werte werden bearbeitet. Es wird nur im Stack gearbeitet.

<u>PS:</u> Ist der Drucker vorhanden, wird zu "SZ" verzweigt. Sonst wird Flag 11 gesetzt, und PRINTER in die Anzeige gebracht. Da zum Einsetzen des Druckersteckers der Rechner ausgeschaltet werden soll (muß!), läuft das Programm nach dem Einschalten automatisch weiter mit

<u>SZ:</u> Im X-Register steht die Adresse der höchsten Speicheradresse. Ist dieser Speicher vorhanden

(Flag 25 bleibt gesetzt) wird ins Hauptprogramm zurückgegangen, sonst wird SIZE nnn im Alpha-Register aufgebaut, wobei nnn die Anzahl der erforderlichen Speicher darstellt und der USER-Modus eingeschaltet. Darauf erfolgt Rücksprung ins Hauptprogramm.

<u>--</u>: Es wird eine Druckzeile mit 12 – in Breitschrift ausgegeben

Beispiel:

```
            PRP "P4K"

 01♦LBL "P4K"
3 XEQ "PS" FC?C 25
PROMPT CF 27 ADV
FIX 2 ADV SF 12
"PLATTE 4 K" PRA CF 12
ADV "GELAGERT" XEQ 02
"R (M):" PROMPT
ASTO 03 ACA ACX PRBUF
"FREI" XEQ 02 CLA
ARCL 03 PROMPT ACA
ACX PRBUF
"(LAGER->RANDMOM"
"⊢ENT" PRA
"M(KNM/M): " PROMPT
STO 03 ARCL X PRA RDN
X<>Y STO 00 / X↑2
STO 01 CHS 1 + 1/X
ST* 03 ADV "R" ACA 8
SKPCHR "MR" ACA 5
SKPCHR "M PHI" ACA
ADV XEQ "—"

 63♦LBL 01
0 "R (M): " PROMPT
X=0? GTO 04 CLA
ARCL X ACA RCL 00 /
X↑2 1/X RCL 01 *
STO 02 CHS 1 +
RCL 03 * 4 XEQ "FT"
RCL 03 1 RCL 02 + *
4 XEQ "FT" ADV GTO 01

 95♦LBL 02
"⊢ER RAND" PRA RTN

 99♦LBL 04
ADV END
```

Es werden 4 Speicher (0 bis 3) benötigt. Sind diese nicht vorhanden ist Flag 25 gelöscht und über PROMPT SIZE 004 angezeigt. Nach Änderung des Speicherbereiches muß mit R/S weitergemacht werden. Der USER-Modus wird abgestellt. Die Schritte 2 bis 6 sind dafür zwingend notwendig.
Schritt 62 druckt die Unterstreichungen. Die erste Zahl der folgenden Zeilen wird mittels ARCL X und ACA in den Druckpuffer geladen.Dadurch kann die erste Stelle ebenfalls besetzt werden (Radien bis 999.99 m). Die folgenden Zahlen können 6 Stellen vor dem Dezimalpunkt haben . (4 XEQ "FT"). Der Druck der Zeile wird durch ADV (Schritt 93) gesteuert.

<u>Beispielausdruck:</u>

```
PLATTE 4 K

GELAGERTER RAND
R (M): 5.00
FREIER RAND
R (M): 15.00
(LAGER->RANDMOMENT
M(KNM/M): 250.00

    R        MR    M PHI
------------------------
  5.00   250.00  -312.50
 10.00    39.06  -101.56
 15.00     0.00   -62.50
```

Dipl. Ing. GEORG RAABE

Liebe Mitglieder !

Ich habe mich sehr über die bisherige Resonanz auf unsere Clubgründung gefreut. So kommt es auch, daß prisma 6-8o sehr umfangreich ausgefallen ist. Außerdem liegt schon Material für prisma 7-8o vor.
Wenn auch Ihnen unser Club gefallen hat, so möchte ich Sie bitten, in den nächsten Wochen dem Mitgliedsbeitrag von 9,--DM für das zweite Halbjahr 1980 zu überweisen. Nach Zahlungseingang erhalten Sie sodann prisma 7-8o. Beachten Sie aber bitte, daß Überweisungen von Banken und Sparkassen bis zu 14 Tagen dauern! prisma 6-8o liegt daher eine Zahlkarte für gebührenfreie Einzahlung bei jedem Postamt bei. Diese ist für Sie besonders dann nützlich, wenn Sie gar kein eigenes Konto besitzen. Weitere Zahlkarten können Sie auf Wunsch erhalten.

OLIVER RIETSCHEL

NOP

NOP kann von einer B2 (Bug 2 -- Seriennummer unter 1951) - Maschine erzeugt werden. Andere 41c ohne diese Möglichkeit können dieses "Programm" beim Club erhalten. Bisher waren nur einige ein- oder zwei-Byte NOP-Befehle bekannt:
1Byte: AOFF (wenn 2 nicht aktiv) ; AON (wenn 2 nicht aktiv) ; DEG ; RAD ; GRAD ; CLD ; LBL oo - 14 ; ADV
2Byte: X<>X ; STO X ; RCL X

Alle diese Befehle können aber Auswirkungen auf den Wert im X-Register haben. Aus der Hex-Code Tabelle des hp-41c geht hervor, daß er die Lücken zwischen Programmschritten mit einem Spare-Code füllt: der Hex-Code ist Fo .
NOP kann nun wie folgt erzeugt werden: 1) Folgendes Programm mit dem Beginn des Speichers eingeben (alle MM herausnehmen):

```
o1  LBL A          o6 ENTER      162  ENTER          167  BEEP
o2  999            o7 ENTER      163  o              168  END
o3  CLA            o8 ENTER      164  LBL E
o4  ARCL IND X        .          165  ASTO IND X
o5  GTO E             :          166  STOP
```

2) PACKEN , USER-Mode setzen
3) A
4) PRGM: 167 BEEP , 1 x "CLEAR" drücken ([←]) und RCL IND T eingeben
5) RUN und E drücken
6) wieder PRGM-Mode: 167 BEEP , 1 x SST drücken. Dann müßte NOP als "" bzw. ᵀ in der Anzeige erscheinen.
7) Die ersten 167 Programmschritte können gelöscht werden und NOP auf eine Kartenseite kopiert werden. Wenn dieser Programmschritt fortan benötigt wird, kann NOP mit "MRG" an das unfertige Programm angehängt werden. Dieses NOP wird im Key-Assingment Programm in den Programmlinien 52 und 99 verwendet.

DETLEV BOCK

Erstellen von Listings für Andere

Ich bin gerne bereit, Programme anderer auszudrucken. Als Kostenbeitrag schlage ich folgende Regelung vor: Pro Zeile o.5 Pfennig, incl. Rückporto, mindestens aber das Rückporto. Der so ermittelte Preis wird auf volle o.5o DM- oder ganze DM-Beträge gerundet. Anschrift: Brotweg 1o ; 332o Salzgitter

Dipl. Ing. GEORG RAABE

Herr Gert-Jürgen Grünler, wohnhaft in 1ooo Berlin 15 , Düsseldorfer Str. 56a , hat sich freundlicherweise bereiterklärt, Programme von Clubmitgliedern gegen einen frankierten Rückumschlag aufzulisten.
Ich persönlich finde, daß es sicher nicht verkehrt wäre, als "kleine Danksagung" eine kurze Programmerklärung beizulegen, sofern die Bedienung des Programms nicht trivial ist.

OLIVER RIETSCHEL

Welche Batterien kaufe ich mir für meinen Rechner? Eine Frage, die sich jeder schon einmal gestellt hat (oder zumindest bald stellen wird).
Brauchen Sie Ihren Rechner nur sehr selten? Dann sind Alkali - Mangan Batterien sicher nicht verkehrt, denn sie sind für langzeitige Belastungen konzipiert. Anders aber bei starker Beanspruchung des Rechners: Hier wird die Anschaffung guter aber billiger Batterien sinnvoll. Sie verkraften vor allem extrem kurzzeitige Belastungen besser als Alkali Batterien. Aus meiner eigenen Erfahrung:
Vor etwa zwei Monaten kaufte ich mir Alkali-Mangan Batterien für meinen Rechner. Am darauffolgenden Sonntag ließ ich etwa 6o - 7o Karten durch den Leser laufen. Das traurige Ergebnis: LOW BAT Daraufhin zog ich es vor, mir billige Batterien zu kaufen und besorgte mir drei Sätze DAIMON N2 Batterien. Doch das Ergebnis war wiederum erstaunlich: Obwohl ich täglich etwa 5-8 Karten durch meinen Rechner jage, hält der erste Satz auch nach zwei Monaten bei ständig angeschlossenem Kartenleser und ein- bis zwei Modulen noch extremsten Belastungen stand.

OLIVER RIETSCHEL

HP User Bibliothek

Jeder, der sich bisher durchgerungen hatte, Mitglied der HP User Bibliothek zu werden, erlebte eine derbe Enttäuschung: Nicht nur, daß alle Programme nur mit englischsprachiger Dokumentation erhältlich sind; nein, mit Ausnahme von zwei oder drei Programmen stammen alle von HP selbst und sind nichts weiter als Kopien aus den überall erhältlichen Solution Books.
Doch soll im Juli endlich ein mehrsprachiger Katalog herauskommen, der auch Programme "von Anwendern für Anwender" enthält. Bleibt nur noch zu hoffen, daß es nicht ähnlich traurig wird, wie beim HP 67/97: Für diese Rechner verzeichnet die User Bibliothek heute genau zwanzig deutschsprachige Programme.

OLIVER RIETSCHEL



Berichtigung:
In prisma 7 muß es in Zeile 274 LBL J heißen, nicht LBL INIT. GÜNTER LELARGE

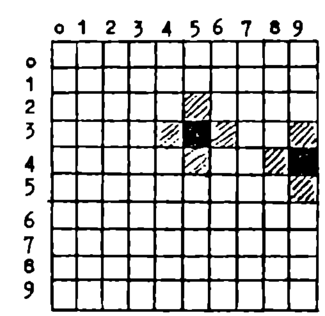
Falschinformation
Mir scheint, als würde HP falsche Auskünfte zu den HP-Programmsammlungen geben:
In Software-Übersichten wird z.B. unter Games angeführt: Hexapawn, Wari, Dice, Nim, others . Aber weder Nim noch Dice sind in der Sammlung enthalten.
OLIVER RIETSCHEL

Funkamateur gesucht !
Befindet sich unter den Mitgliedern ein lizensierter Funkamateur, der vielleicht an einem Programm arbeitet, das durch Eingabe der QTH - Kennern (meine QTH-Kenner ist GJ 11 h) die Entfernung in Kilometern errechnen kann ? Ich bin selbst Funkamateur mit dem Rufzeichen " DL 8 KE " . RUDOLF EBER

U-Boot-Jagd
Jedem kann einmal ein Fehler unterlaufen. Auch wenn das keine richtige Entschuldigung ist, so hoffe ich doch, Sie werden es mir nachsehen: Ganz zu Anfang unseres Clubbestehens habe ich unglücklicherweise vier funktionsuntüchtige Versionen eines von mir entworfenen Spiels "U-Boot-Jagd" versandt. Ich habe das Spiel nun überholt und sehe drei Gründe dafür, es in prisma abzudrucken:

1) Wiedergutmachung für die erste Version
2) Meiner Meinung nach kann das Programm als Beispiel dafür gelten, daß auch Mammutprogramme bei entsprechender alphanumerischer Auslegung durchaus übersichtlich sein können.
3) Der Vorschlag von Günter Lelarge, alle Programme abzudrucken, stieß sowohl auf Zustimmung als auch auf Ablehnung, so daß ich im weiteren bei Beibehaltung des Programmkataloges auch verstärkt Programme in prisma bringen werde.

Hier nun eine kurze Beschreibung des Spiels:



Gespielt wird auf einem 1o x 1o Feld mit 5 U-Booten. Sowohl der Rechner als auch der Spieler verstecken vor Spielbeginn 5 U-Boote auf 5 von diesen Feldern. Der Rechner braucht hierzu die Eingabe einer Zahl z mit o<z<1 . In der Anzeige erscheint dann das Feld, auf das der Rechner schießt. Liegt auf diesem Feld eines Ihrer U-Boote, so hat der Rechner einen Treffer erzielt und Sie müssen die Taste A drücken. Ein Fast-Treffer (Taste B) liegt dann vor, wenn auf einem der unmittelbar angrenzenden Felder eines Ihrer U-Boote liegt. Beispiel: U-Boot auf 35: Fast Treffer bei Schuß auf Feld: 25,34,36,45
U-Boot auf Feld 49: Fast-Treffer bei Schuß auf Feld: 39,48,59, aber nicht bei Schuß auf Feld 5o ! (Randlage)
Liegt weder ein Fast-Treffer noch ein Treffer vor, so drücken Sie die Taste C (Fehlschuß). Daraufhin sind Sie mit Schießen an der Reihe. Geben Sie das Feld ein, das Sie zu beschießen wünschen und der Rechner wird über Erfolg oder Mißerfolg Auskunft geben. Die Rechenzeit variiert, besonders in der Schlußphase des Spiels, mit den Problemen, vor die sich der Rechner gestellt sieht. Sollte er nach mehreren Minuten noch keine Antwort geben, so hat er bereits alle Felder beschossen und Sie haben ihm ein U-Boot "unterschlagen" . In diversen Probespielen erwies sich das Spiel als fehlerfrei und spielstark. "Fehler" waren immer auf Fehler im Festhalten der Spielgeschehnisse zurückzuführen. Notieren Sie sich jeden Schuß genauestens! Viel Spaß!
OLIVER RIETSCHEL

PS: Das Programm benötigt zwei Memory-Module, doch läßt es sich durch alphanumerische Kürzungen auch mit einem Memory-Modul ausführen.

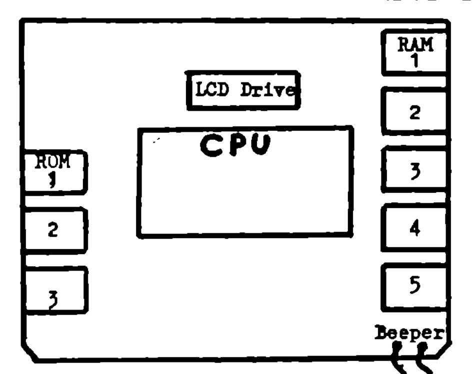
Aufgrund des enormen Umfangs von prisma 7-8o wird das nächste prisma (9-8o) erst in ca. 8 Wochen erscheinen. Eine andere Regelung ist bei 1,5oDM monatlich leider nicht möglich .

Danke!

Im folgenden finden Sie den Bericht "Rechnerorganisation des HP-41C" . Die Informationen hierzu habe ich aus den verschiedensten Quellen erhalten. Ich möchte mich dafür bei allen Informanten für die Telefonanrufe und Briefe auch im Namen des Clubs bedanken. Besonders hervorheben möchte ich Detlev Bock, Göttingen, HP-Vertriebsleitung, Frankfurt, und verschiedene Berichte in den Ausgaben von PPC Journal, Santa Ana, aus den letzten 1o Monaten, die mir alle ein wenig weiterhalfen oder meine Erkenntnisse bestätigten. OLIVER RIETSCHEL

Rechnerorganisation des HP-41C

Was steckt in unseren Rechnern ? Eine kurze Frage und eine kurze Antwort ? Gewiß nicht ! Wahrscheinlich haben nur sehr wenige von uns bisher ihren Rechner "aufgerissen", denn das ist auch nicht zu empfehlen: Garantieverlust, extreme Empfindlichkeit durch CMOS Technologie, hohe Reparaturkosten -- gewichtige Gründe also. Darum sei die Anleitung zum Auseinanderbau auch erst einmal verschwiegen - es lohnt sich nicht für den Elektroniklaien. Hier nun eine Skizze der Logik-Platine:

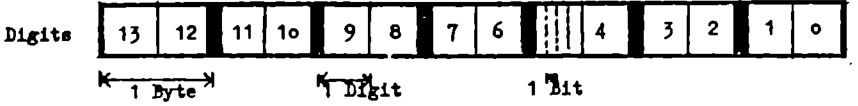


Unschwer zu erkennen: Die CPU (Central Processor Unit), die zentrale Recheneinheit in der Mitte. Sie ist der eigentliche Rechner und steuert die Speicher an. Von den Speichern sind da zunächst einmal 12kByte ROM (Read Only Memory)(3 Chips mit je 4kByte), das Betriebssystem, das auch ohne Stromversorgung nicht zerstört wird. Weiterhin findet man noch fünf weitere Chips: Es handelt sich um RAM's (Schreib- Lese- Speicher), die Programmdaten und Speicherregister beinhalten; jedes RAM mit einer Kapazität von 112 Byte. 112 Byte entsprechen 16 Registern. Bei 5 Chips also 8o Register. Davon sind uns allen 64 Register bestens bekannt: Der frei programmierbare RAM. Weitere RAM Kapazität läßt sich über die I/O - Ports anschließen: Memory Module mit je 448 Byte, maximal also 1792 Byte. Das bedeutet, daß der HP-41C über bis zu 2,352kByte RAM und 12kByte ROM verfügt. Außerdem lassen sich weiterhin bis zu 48kByte ROM an die I/O-Ports anschließen.

Doch zurück zur Standardversion: Von den 5 RAM sind alsp 4 der frei zugängliche Speicherplatz, über dessen Gestaltung und Aufbau sicher noch ein eigener Bericht folgt. Doch was ist mit dem fünften ? Eine Frage, die nur durch extrem aufwendiges Probieren geknackt werden konnte. Man kann auch "von hinten herum" kommen: Was beinhaltet die erste Seite der Statuskarten? Nun, man kann es ausprobieren: Zunächst sind da einmal die XYZTL - Register, ferner das Alpharegister und die Flags. Doch eingehendere Untersuchungen amerikanischer "Softwaretechniker" ergaben, daß es sich hierbei vielmehr um einen wohl definierten Block von 16 Registern handelt. HP bestätigte mir dies.

Doch zuvor ein klärendes Wort: 1kByte = 1oooByte ; 1 Register = 7 Bytes = 14 Digits
1 Digit = 4 Bits = 1 hexadezimales Zeichen

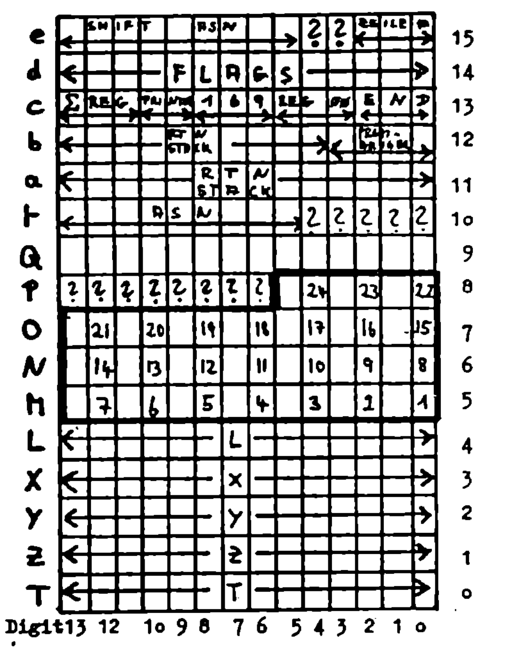
Ein typisches HP-41C Register sieht also so aus:



Werden nun numerische Werte in ein Register abgespeichert, nehmen die Digits die folgenden Werte an:

Digit 13: o für positive Werte , 9 für negative Werte
Digit 12-3: 1ostellige Mantisse des abgespeicherten Wertes
Digit 2 : o für positiven Exponenten , ≠ o für negativen Exp.
Digit 1-o : 2stelliger Exponent des abgespeicherten Wertes

Doch zurück zu unserem 16 Register Block:



Benennen wir die Register erst einmal mit den Zahlen o bis 15:
Im Register o finden wir das T Register.
Im Register 1 das Z, in 2 das Y, in 3 das X und in 4 das Last X Register.
Register 5, 6 & 7 und die ersten drei Byt von Register 8 enthalten das Alpharegiste. Diese Register stellen sich in der Anzeig als Register M, N, O & P dar. Vergleichen Sie prisma 2 & 3 ! Die restlichen 4 Bytes von Register 8 (P) enthalten möglicherweise (??) nur während des Beschreibens und Lesens von Magnetkarten die SIZE und ΣREG Zuordnung.
Register9 (in der Anzeige: Q) enthält, wenn kein Drucker angeschlossen ist, den umgekehrten Hexcode des zuletzt angesprun genen Alphalabels.
Register 1o (Anzeige: ŀ) enthält die Tastenzuordnungen ohne SHIFT. Ebenso wie in Register 15 werden aber nur Standardfunktionszuordnungen vermerkt. Welche Informationen die verbleibenden fünf Digits be inhalten ist mir z.Z. noch unklar.
Die beiden folgenden Register 11 & 12 stellen sich in der Anzeige als Register a & b dar. Sie enthalten die sechs Rücksprungadressen, die sich der Rechner bei sechs Unterprogrammebenen merken muß. Die einzenen Adressen werden in jeweils 2 Bytes unterteilt und zwar in folgender Form:

AXYZ XYZ ist hierbei der Hexcode des Registers, in das der Rechner zurückspringe muß,
A ist das Byte, bei dem die Programmausführung fortgesetzt wird.

Register b enthält außerdem noch den Programmzeiger, der angibt, in welchem Byte der Rechner sich gerade befindet.
Im Register 13 (c) findet sich schließlich eine ganze Anzahl verschiedener Informationen: Die ersten drei Digits enthalten die .END. Marke im Programmspeicher. Die nächsten drei Digits enthalten die Position des ersten Speicherregisters im RAM und die darauf folgenden drei Digits sind besonders interessant: Der Rechner tut hier einen ganz bestimmten Wert hinein (bei meinem Rechner: 169) und kontrolliert diesen dann ständig. Sobald sich dieser Wert in irgendeiner Form verändert, wird angenommen, daß auch der restlichen RAM in Mitleidenschaft gezogen wurde und der Rechner löscht selbsttätig den gesamten RAM und gibt die Meldung MEMORY LOST aus. Weiterhin folgen in Register c zwei Digits, die die Status-Flags des Druckers beinhalten und drei Digits, in denen die Adresse des ersten Summenregisters zu finden ist. Alles selbstverständlich hexadezimal.
Im Register 14 (d) finden wir die Flags des Rechners: 1 Register entspricht 56 Bit und 1 Bit entspricht 1 Flag. Aus diesem Grund hat der HP-41C auch 56 Flags.
Im Register 15 finden wir die Programmzeile, in der der Rechner sich im Moment gerade befindet, in den ersten drei Digits. Dann folgt 1 Byte, dessen Verwendung mir momentan noch unklar ist und neun Digits, die die Tastenzuordnungen mit SHIFT enthalten. Wie in Register 1o wird hier nicht verzeichnet, was der Taste zugeordnet ist, sondern ob ihr etwas zugeordnet ist.
Wir wollen die gefundenen Register in Zukunft so bezeichnen, wie sie uns die Anzeige bietet: Als Register T,Z,Y,X,L,M,N,O,P,Q,ŀ,a,b,c,d,e
Druckerbesitzer werden bemerken: In den Listings werden einige dieser Register anders dargestellt: Sie sehen dort z.B. so aus: RCL ŀ als RCL ⊤ ; RCL M als RCl RCL N als RCL \ (Siehe auch Programm Key-Assingment)

In prisma 2 haben wir weiterhin die Registerbefehle zu den Registern A,B,C,D erzeugt. Wo sind die denn nun ? Erinnern wir uns, daß direkt nur Werte in die Speicherregister oo bis 99 getan werden können, da der Cursor eine zweistellige Adresse verlangt. Bei meinem Rechner ist es nun so, daß durch die Registerbefehle zum Register A das Speicherregister 1o2 angesprochen wird. Wenn auch etwas umständlich, so ist damit doch ein direktes Benutzen der Register 1o2 bis 111 (und noch weiter???) möglich, wenn man die Registerbefehle der Register A,B,C,D,E,F,G,H,I,J benutzt, die wahrscheinlich alle existieren.

Wie numerische Werte in Speicherregister abgespeichert werden, habe ich oben bereits erklärt . Dooh wie ist es nun, wenn der RAM als Programmspeicher benutzt wird?
Die Einteilung in Register entfällt, wichtig wird die Einteilung in Bytes oder Digits
Erinnern wir uns: Ein Digit ist ein hexadezimales Zeichen, d.h. eine 4-bit-Dualzahl.
HP hat das Problem nun folgendermaßen gelöst: Ein Register wird in 7 Bytes unterteilt (s.o.). Jedes Byte kann, da es zwei Digits enthält, 16 x 16 = 256 verschiedene Werte annehmen. Bei der Verarbeitung eines Programmes geschieht nun folgendes:
Die CPU "betrachtet" nacheinander die einzelnen Bytes oder besser: deren Inhalte.
Jeder einzelne dieser 256 verschiedenen Codes hat eine ganz bestimmte Bedeutung.
Sie läßt sich in der sogenannten Hexcodetabelle des HP-41C nachsehen. Sie enthält zunächst einmal alle Standardfunktionen des Rechners, geordnet nach einem bestimmten Ziel: Gleichartige Funktionen in Blöcken zusammenzufassen. Hierbei spielen z.B. die folgenden Gesichtspunkte eine Rolle: Wieviel Variablen werden durch die Funktion verändert? Wieviel Gemeinsamkeiten weisen die einzelnen Verarbeitungsroutinen auf? Welche Register werden benutzt? u.s.w.
Hierbei spielen vor allem elektronische und organisatorische Probleme eine Rolle, deren Erörterung langwierig und für viele von uns wenig interessant wäre. Deshalb will ich hier nicht näher darauf eingehen. Wichtig ist jedoch noch:
Nicht alle 256 Codes konnten (und sollten !!!) belegt werden: Denn für die häufig verwendeten Befehle, wie RCL, STO, GTO , LBL wurden für die ersten 15 bzw. 16 Befehle eigene Codes eingeplant: Ein ausgezeichneter Einfall, denn dadurch wird der durch ein Programm benötigte Speicherplatz deutlich geringer. Die übrigen Befehle brauchen nämlich je zwei Bytes: eines zur Festlegung der Funktion, eins zur Festlegung der Adresse. Sehen Sie sich die Hexcodetabelle ruhig näher an, Sie werden sicher noch einiges interessantes entdecken:

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NULL	LBL 00	01	02	03	04	05	06	07	08	09	10	11	12	13	14
1	0	1	2	3	4	5	6	7	8	9	.	EEX	CHS	GTO•	XEQ•	SPARE
2	RCL 00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
3	STO 00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
4	+	-	*	/	x<y?	x>y?	x≤y?	Σ+	Σ-	HMS+	HMS-	MOD	%	%CH	P-R	R-P
5	LN	x^2	$\sqrt{x}$	y^x	CHS	e^x	LOG	10^x	e^x-1	sin	cos	tan	asin	acos	atan	→DEC
6	1/x	ABS	FACT	x≠0?	x>0?	LN1+x	x<0?	x=0?	INT	FRAC	D-R	R-D	→HMS	→HR	RND	→OCT
7	CLΣ	x<>y	PI	CLST	R↑	RDN	LASTX	CLX	x=y?	x≠y?	SIGN	x≤0?	MEAN	SDEV	AVIEW	CLD
8	DEG	RAD	GRAD	ENTER↑	STOP	RTN	BEEP	CLA	ASHF	PSE	CLRG	AOFF	AON	OFF	PROMPT	ADV
9	RCL	STO	STO+	STO-	STO*	STO/	ISG	DSE	VIEW	ΣREG	ASTO	ARCL	FIX	SCI	ENG	TONE
A	XROM	→							SF	CF	FS?C	FC?C	FS?	FC?	XGI	SPARE
B	SPARE	GTO 00	01	02	03	04	05	06	07	08	09	10	11	12	13	14
C	ALBL	END	→												x<>	LBL nn
D	GTO	→														
E	XEQ	→														
F		TEXT 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Zeilen o bis 8: 1 Byte Befehle ; Zeilen 9 bis C: 2 Byte ; Zeilen D & E: 3 Byte

Ein kurzes Beispiel: In einem Register seien die folgenden Werte abgespeichert:



LBL ØØ   4   *   SQRT   STO   74   RTN

Haben Sie sich die Tabelle genau angesehen ?
Dann haben Sie sicher noch einige seltsame Codes entdeckt !

Spare ; Code: 1F und AF und Bo (unf Fo ???)
Dieser Code hat keine Wirkung auf den Rechner, befindet sich vor PACKING zwischen den Programmzeilen und ermöglicht dadurch schnelles Einfügen weiterer Befehle. Die Unterschiede zwischen den einzelnen Codes sind mir im Moment noch nicht bekannt.

Null ; Code: oo
Wird ein Befehl im Programmspeicher gelöscht, so wird in diese(s) Byte(s) eine o getan, das Byte wird gelöscht. Erst die Funktion PACK entscheidet, ob der NULL Code hier entbehrlich ist. Er ist z.B. nicht entbehrlich, wenn durch ihn zwei numerische Werte getrennt werden: Das Löschen von ENTER bringt in folgender Routine auch nach PACKING keinen Speicherplatzgewinn, da anstelle von ENTER hier jetzt der ("unsichtbare") NULL Code steht:

```
  :                          :
  19                         19
ENTER      nach PACKING:    NULL
  3            + [←]          3
  :                          :
```

Das längere Drücken einer Taste führt zur automatischen Löschung der Funktion: Es wird statt dessen der NULL Befehl ausgeführt.

XGI ; Code: AE
XGI entspricht ...IND...

Doch jetzt erst einmal genug Theorie: Gibt es auch irgendeinen praktischen Nutzen der ganzen Sache ? Gewiß !
Hier einige Anregungen:

Register:	
M,N,O,P	Manipulation im Alpharegister ; zusätzliche Speicher
a,b	zusätzliche Speicher ; Manipulation der Rücksprungadressen
c	programmierbares MEMORY LOST (durch STO c)
d	gezieltes Löschen oder Setzen größerer Gruppen von Flags. Manipulation bisher nicht beeinflußbarer Flags.
e	Springen zu einer bestimmten Programmzeile (ohne LBL Befehl) Löschen aller oder Teile der Tastenzuordnungen (auch: Register ⊦

Sicherlich gibt es noch viele Anwendungsmöglichkeiten mehr, doch heißt es zuvor:
P r o b i e r e n !
Meine Bitte an alle Experimentatoren: Teilen Sie mir mit, was für Anwendungsmöglichkeiten Sie entdeckt haben !
Doch halt: Wie kommen Sie an die Befehle ?? Ganz einfach: Schicken Sie mir zwei Magnetkarten, Sie bekommen die Registerbefehle dann als Tastenzuordnungen zugeschick (Statuskarte). Oder - wenn Sie das Programm Key-Assingment schon besitzen - probier probieren, denn die Befehle lassen sich damit erzeugen. Doch wie ? Noch weiß ich es auch nicht so genau - also: vielleicht im nächsten prisma ??! Viel Erfolg !

OLIVER RIETSCHEL

Re - Assignment - Programm
1) Master Clear ausführen
2) Die Funktion TAN der eigenen Taste zuordnen
3) WSTS: WSTS verlangt zwei Seiten: In diesem Falle wird aber nur eine Seite benötigt, so daß dieselbe Kartenseite 2 x durchlaufen muß. Auf der anderen Seite können Sie z.B. das NOP speichern.

Der Gebrauch der Karte ist sehr einfach: statt den Rechner mit MASTER CLEAR gewalttätig zu löschen, wird die Karte eingelesen, dann auf [←] gedrückt und nur noch TAN von seiner Taste gelöst. Der Rechner hat dann wieder seine alte Registeranzahl zur Verfügung. Der Vorgang ist praktisch der gleiche als wenn man jede einzelne Tastenzuordnung mit ASN "" löschen würde.

DETLEV BOCK

```
01•LBL "KA"
FIX 0 "PAIRS?" PROMPT
1 - 1 E3 / STO 02
CLX STO 04 XEQ B 1
XEQ A 105 XEQ A 12
XEQ A 0 XEQ A 191
XEQ A ASTO 05

 24•LBL 06
XEQ B 240 XEQ A
XEQ 07 XEQ 07 RCL 02
RCL [ RCL c RCL 05
STO c X<> Z STO IND T
RDN STO c ISG 02
GTO 06 CLX BEEP RTN

 44•LBL C
"MSTS" AVIEW PSE MSTS
RTN

 50•LBL 07
ISG 04 "" "KEY "
ARCL 04 TONE 9 PROMPT
STO 01 RDN STO 03 RDN
XEQ A -24 X<> 03
XEQ A 10 ST/ 01
RCL 01 ABS INT LASTX
FRC ST* Z .1 - X=0?
GTO 05 RDN 4 X=Y?
ISG Z

 81•LBL 05
RDN STO Z X<>Y 8
X<>Y ST* Y RDN + CHS
44 + ST+ 03 30 X>Y?
SF 00 RDN DSE Z ""
X<> 01 X<0? GTO 01
RCL ' XEQ 03 STO '
GTO 08

107•LBL 01
8 ST+ Z RDN RCL e
XEQ 03 STO e

114•LBL 06
RDN RDN X<>Y 16 * +

121•LBL A
16 X<>Y 0 X<> d X<>Y
SF 03

128•LBL 04
2 / FRC DSE Z X≠0?
SF IND Z X<> L INT
X≠0? GTO 04 X<>Y
X<> d "******" ARCL X
ASTO X "*" ARCL X
ASHF ASTO X CLA
ASTO Y ARCL 00 ARCL X
ASTO 00 X≠Y? RTN
RCL [ STO \ "├*"
RCL \ CLA STO [
ASTO 00 RTN

163•LBL 03
FS?C 00 GTO 09 CLA
X<> \ "├***" X<> \
X<> d SF IND 03 X<> d
X<> \ "├****" X<> ]
RTN

177•LBL 09
X<> d SF IND 01 X<> d
RTN

182•LBL B
CLA ASTO 00 END
```

Key - Assignment - Programm

Mit diesem Programm ist es nicht nur möglich, alle Befehle einer beliebigen Taste zuzuordnen, sondern es lassen sich auch weitere HP-41C Funktionen erzeugen. Es werden die Registerbefehle der Register M,N,├, c und d benötigt. Sie können das Programm auf zwei Magnetkarten erhalten oder selbst eingeben, sofern Sie die nötigen Statuskarten bereits besitzen.

Benutzung des Programms:
Das Programm wird mit XEQ "KA" gestartet. Die Anzahl zuzordnender Funktionen wird abgefragt und dann die Zuordnung selbst. Leider ist der ISG o2 Befehl nicht richtig eingesetzt worden. Die Abfrage "PAIRS?" könnte daher weggelassen werden. Das Übertragen auf Karte kann selbst mit USER C vorgenommen werden. Das Programm stammt von Tom Cadwallader aus PPC Calculator Journal April 8o.

Auf die Abfrage "KEY?" werden 3 durch ENTER getrennte numerische Werte eingegeben: Die ersten beiden Werte sind aus der Hexcode-Tabelle zu entnehmen: 159 = TONE , da der 159. in der Tabelle verzeichnete Befehl TONE heißt. Ebenso: RCL = 144 ; ST+ = 146 ; J = 111 ; T = 112 . Der dritte Wert gibt die Taste an, der etwas zugeordnet werden soll: XY: Die Yte Taste in der Xten Reihe.
z.B.: 14 = LOG . Als Beispiele, was sich so alles machen läßt, seien hier kurz genannt:

4 ENTER 167 ENTER XY (Taste)	gibt:	eGØBEEP _ _
4 ENTER 162 ENTER XY	gibt:	ØBEEP _
4 ENTER 161 ENTER XY	gibt:	⊬ _ _
4 ENTER 16o ENTER XY	gibt:	↗ _ _
159 ENTER 26 ENTER XY	gibt:	TONE 26
168 ENTER 49 ENTER XY	gibt:	SF 49 (funktioniert nur bei einigen Rechnern)
144 ENTER 111 ENTER XY	gibt:	RCL J

eGØBEEP & ØBEEP verlangen numerische bzw. alphanumerische Eingaben. Damit ist es möglich, Programme in den Peripherieeinheiten anzusprechen, ohne diese zu besitzen (bzw. nicht angeschlossen zu haben) Der XROM sieht so aus: XROM nn,mm ; nn=Peripherieeinheit, mm=oo bis 64 ist der Code für die Programme. NONEXISTENT erscheint im RUN-Mode, wenn die nn Einheit nicht angeschlossen ist. Ohne Probleme kann man im PRGM-Mode die Befehle überprüfen. Beispiele: eGØBEEP nn= oo gibt XROM 28,oo ; nn= 63 gibt XROM 28,63 ; nn= 64 bis 99 gibt XROM 29,oo bis 29,35 (Drucker !)
↗nn: nn= 3o gibt 7ISZI (Kartenleser); nn= 64 bis 99 gibt MATH 1A bis TANH (Mathemodul).
↗oo läßt den Rechner sterben, hat aber erstaunliche Folgen. Probieren Sie es aus (wenn kurzes Herausnehmen der Batterien nichts mehr hilft, Batterien ein paar Minuten rausnehmen) Der Rechner kommt ohne Defekte wieder. Er kann auch in den SLEEP-Mode abrutschen und kommt dann nach kurzer Zeit selbst wieder zurück.

DETLEV BOCK

Brettspiele für HP-41C

Wer arbeitet an Programmen zu Brettspielen oder hat schon welche erstellt? Ich selbst "tüftle" derzeit an "Reversi", ohne jedoch bisher eine erfolgreiche Strategie gefunden zu haben. Aber auch Dame, Mühle etc. habe ich ins Auge gefaßt. Ich würde mich freuen, wenn darüber ein Ideen- und Erfahrungsaustausch zustandekäme.

ULRICH DAVERTZHOFEN
Hünefeldstr. 1o9 ; 56oo Wuppertal 2

```
PRP "

01♦LBL "U-BOOT"
"O. RIETSCHEL"
"POSTF. 373"
"D-2420 EUTIN"
"30. 05. 80" CLRG
FIX 0 SF 27 CF 29
XEQ "FLAGS 2" "0<X<1"
PROMPT STO 00 5 E-3
STO 06

16♦LBL 99
RCL 00 9821 * .211327
+ FRC 1.1111 * FRC
STO 00 1 E2 * INT
1.005 STO 07 X<>Y

33♦LBL 98
RCL IND 07 X<>Y X=Y?
GTO 99 ISG 07 GTO 98
STO IND 06 ISG 06
GTO 99 RCL 00 STO 21

45♦LBL "SCHUSS"
FC? 00 GTO "NEU"
XEQ "FLAGS" RCL 06
XEQ "NACHBAR" 1 STO 08
FS?C 16 FC? 01 GTO 02
RCL 06 10 - STO 00
GTO 06

61♦LBL 02
FS?C 17 FC? 02 GTO 03
RCL 06 1 - STO 00
GTO 06

70♦LBL 03
FS?C 18 FC? 03 GTO 04
RCL 06 10 + STO 00
GTO 06

79♦LBL 04
FS?C 19 FC? 04
GTO "SCHUMM" RCL 06 1
+ STO 00

87♦LBL 06
RCL 00 XEQ "ABFRAGE"
X=0? GTO 07 FC? 00
GTO "NEU" 1 ST+ 08
GTO IND 08

97♦LBL 05
GTO "SCHUMM"

99♦LBL 07
RCL 00 XEQ "FREI"
" ** " ASTO L ARCL 00
ARCL L RCL 00 PROMPT

108♦LBL "NEU"
26.05 ST+ 21 RCL 21
INT STO 00 1 E2 X>Y?
GTO 06 ST- 21 ST- 00
GTO 06

120♦LBL "FREI"
XEQ "UP1" ST+ IND Y
RTN

124♦LBL "ABFRAGE"
STO 00 XEQ "UP1"
RCL IND Y X<>Y / INT
10 / FRC 10 * RTN

137♦LBL "UP1"
.1 * INT LASTX FRC
10 * 10↑X X<>Y 10 +
X<>Y RTN

151♦LBL "NACHBAR"
10 - X<0? CF 01 80
X<=Y? CF 03 RDN 10 /
FRC 10 * 0 X=Y?
CF 02 X<>Y 9 X=Y?
CF 04 RTN

173♦LBL "FLAGS"
SF 01 SF 02 SF 03
SF 04 RTN

179♦LBL "FLAGS 2"
SF 16 SF 17 SF 18
SF 19 CF 00 RTN

186♦LBL A
XEQ "FLAGS 2" FS?C 05
SF 00 RCL 07 STO 06 1
ST+ 20 RCL 20 5 X=Y?
GTO "HP" GTO "SPIELER"

199♦LBL B
FS? 00 SF 05 SF 00
FS? 05 STO 07 FC? 05
STO 06 GTO "SPIELER"

208♦LBL C
XEQ "FLAGS" RCL 00
XEQ "NACHBAR" RCL 00
10 - FS? 01
XEQ "FREI" RCL 00 1 -
FS? 02 XEQ "FREI"
RCL 00 10 + FS? 03
XEQ "FREI" RCL 00 1 +
FS? 04 XEQ "FREI"

232♦LBL "SPIELER"
"IHR SCHUSS ?" PROMPT
CLA STO 00 XEQ 21 PI
CHS FS? 09 STO IND 08
SF 00 FS?C 09
GTO "TREFFER" CF 00
XEQ "FLAGS" RCL 00
XEQ "NACHBAR" 10
ST- 00 FS? 01 XEQ 21
9 ST+ 00 FS? 02
XEQ 21 2 ST÷ 03
FS? 04 XEQ 21 9
ST+ 00 FS? 05 XEQ 21
FS?C 09 GTO "FAST"
"FEHLSCHUSS" AVIEW PSE
GTO "SCHUSS"

271♦LBL 21
1.005 STO 08

274♦LBL 97
RCL 00 RCL IND 08 X=Y?
SF 09 X=Y? RTN ISG 08
GTO 97 RTN

284♦LBL "FAST"
"FAST-"

286♦LBL "TREFFER"
"⊢TREFFER" AVIEW BEEP
PSE 0 FS?C 03 1
ST+ 09 RCL 09 5 X=Y?
GTO "SIEG" GTO "SCHUSS"

300♦LBL "HP"
"** HP-41C **" PROMPT

303♦LBL "SIEG"
" GRATULIERE" PROMPT

306♦LBL "SCHUMM"
BEEP " SCHUMMLER"
AVIEW PSE GTO "HP"
.END.
```



Preise? Preise!

Für Clubmitglieder konnte ein besonderer Service errungen werden: Auf alle HP-Preise erhalten Sie jetzt 1o% Rabatt ! Außerdem ist fast alles zu sofort oder wenigstens kurzfristig lieferbar. Für Druckerbesitzer besonders interessant: Schwarzdruckendes Papier für den 41C Drucker: Vorteile: Verbleicht im Gegensatz zum HP Papier kaum. Wesentlich billiger. Möglicher Nachteil: HP verweigert, was bei den HP Preisen auch verständlich ist, jede Auskunft darüber, wie der Druckkopf des Thermodruckers reagiert: Sinkt die Lebensdauer von - sagen wir mal - 5oo auf 4oo oder gar nur 3oo Rollen ? Oder erhöht sie sich auf weit über 5oo Rollen ? Beides ist möglich, doch sind mit diesem Papier noch keine negativen Erfahrungen gemacht worden. Dennoch bleibt ein Restrisiko unter dem Gesichtspunkt, daß auch ein neuer Druckkopf billiger wäre, als der Mehrpreis für nur 5o HP-Rollen. Der Club-Preis für 1 Rolle Thermopapier: 1,2o DM! Bei weiteren Fragen und Bestellungen wenden Sie sich bitte an:
GÜNTER LELARGE ; TEL.: o2636/2137

Copyright ?

Alle in prisma erschienenen Artikel dürfen Sie in beliebiger Form weiterverwenden, auch kopieren. Sie sollten dabei abar im Interesse des Clubs handeln und immer den Autor angeben.

MALFUNCTION

Für alle die es noch nicht wußten: Läßt sich eine Magnetkarte nicht einlesen und es wird MALFUNCTION ausgegeben, so ist die Karte verdreckt. Eine Kleinigkeit kann die Ursache sein - und die Abhilfe: Die Karte mit einem in Alkohol getränkten Papiertaschentuch abwischen, fertig! Es hilft fast immer.

Rennfahrer-Spiel für HP-41C

Beschreibung: Der Spieler hat mit dem HP-41C einen Rennwagen in der Hand (wahlweise auch ein Mtorrad), mit dem er möglichst schnell eine bestimmte Strecke zurücklegen muß. Dabei muß er sich allerdings an gewisse Tempolimits halten, da ihm sonst Strafsekunden drohen. Er kann Gas geben, bremsen und Gänge einlegen. Am Ende der Strecke wird die gefahrene Zeit angezeigt, so können mehrere Spieler versuchen, Rundenrekorde etc. aufzustellen. (kein Memory nötig!!!)
Länge: 2o3 Zeilen ; PG: III ; Entwicklung: Edmund Weitz ; 2 Magnetkarten
Bei Verrechnung mit Programmgutscheinen: Wertigkeit 4oo Zeilen

Anmerkung: Das Programm, allein durch seine 8seitige Dokumentation beeindruckend, ist wirklich interessant, gut zu bedienen, wird nicht so schnell langweilig und gehört allein von der Idee her mit in die vorderste Linie der guten Spielprogramme. In Zeile 2 würde ich einfügen: CLRG oder zumindest: o STO o5 .
Dem Autor meinen herzlichen Glückwunsch ! OLIVER RIETSCHEL

Tips für Anfänger

Beim Arbeiten mit dem Handbuch kam Harald Krumins aus Eckernförde eine Idee:
Er schrieb folgende Routine zum Anzeigen von Text bzw. Text mit Ergebnis:

```
LBL"L"
AVIEW
PSE
END
LBL"M"
ARCL X
AVIEW
PSE
TONE
STOP
END
```

Seine Programme versah er jetzt mit kurzen abstrakten Labels, gefolgt vom Programmnamen und führte dann "L" aus. Zur Ergebnisanzeige verwendete er "M" . Beispiel:

```
LBL"AA"       LBL"BB
"VOLUMEN"     "OBERFLÄCHE"
XEQ"L"        XEQ"L"
XEQ"BB"       "RADIUS ?"
RCL oo        PROMPT
x             STO oo
3             X↑2
/             PI
"V : "        x
XEQ"M"        4        "A : "
END           x        XEQ"M"
                       END
```

Ein netter Einfall, doch ist zumindest LBL"L" in der Praxis uneffektiv, denn XEQ"L" ist zwar optisch kürzer als AVIEW PSE , braucht aber 3 Bytes Speicherplatz. Die Befehle AVIEW & PSE brauchen zusammen aber nur 2 Bytes. Sicherlich ein guter Hinweis für alle "Anfänger", die nur nach der optischen Programmlänge gehen; oder wußten Sie "alter Hase" das auch noch nicht ? LBL"M" dagegen ist bei dem hier geforderten Zweck nicht weiter zu verkürzen. OLIVER RIETSCHEL

(S.192 Nr.2)

MEMORY

```
01◆LBL "MEM"
02 RCL 00
03 CLRG
04 STO 00
05 FIX 0
06 CF 29
07 1.01
08 STO 21
09◆LBL 01
10 CLA
11 ASTO X
12 9999
13 XEQ 00
14 STO [
15◆LBL 03
16 XEQ 02
17 RCL IND X
18 SF 25
19 X<=Y?
20 DEG
21 FC? 25
22 GTO 03
23 ASTO IND Y
24◆LBL 04
25 XEQ 02
26 RCL IND X
27 SF 25
28 X<=Y?
29 DEG
30 FC? 25
31 GTO 04
32 ASTO IND Y
33 ISG 21
34 GTO 01
35 0
36 STO 21
37◆LBL A
38 "BITTE RATEN"
39 AVIEW
40 TONE 0
41 PSE
42 CLA
43 XEQ 05
44 "⊢1 U. 2?"
45 PROMPT
46 RCL IND Y
47 PSE
48 PSE
49 RCL IND Y
50 PSE
51 PSE
52 RDN
53 RDN
54 X=Y?
55 GTO A
56 RDN
57 RDN
58 X≠Y?
59 GTO A
60 "LEER"
61 ASTO X
62 X=Y?
63 GTO A
64 ASTO IND T
65 ASTO IND Z
66 "IHRE "
67 XEQ 05
68 "⊢?"
69 PROMPT
70 20
71 +
72 1
73 ST+ IND Y
74 RCL 21
75 RCL 22
76 +
77 10
78 X≠Y?
79 GTO A
80 "ENDE"
81 AVIEW
82 BEEP
83 XEQ 06
84 "⊢1="
85 ARCL 21
86 AVIEW
87 PSE
88 XEQ 06
89 "⊢2="
90 ARCL 22
91 AVIEW
92 XEQ 06
93 RCL 21
94 RCL 22
95 X>Y?
96 "⊢2"
97 X<Y?
98 "⊢1"
99 X=Y?
100 "GLEICH"
101 AVIEW
102 BEEP
103 BEEP
104 RTN
105◆LBL 06
106 "SPIELER"
107 RTN
108◆LBL 05
109 "⊢NUMMER"
110 RTN
111◆LBL 02
112 20
113 XEQ 00
114 1
115 +
116 RTN
117◆LBL 00
118 RCL 00
119 9821
120 *
121 .211327
122 +
123 FRC
124 STO 00
125 *
126 INT
127 END
```

ZEILE 14 HEISST: STO M

<u>Memory-Spiel für den HP-41C von Edmund Weitz</u>

Kurzbeschreibung:
Das Spiel ähnelt dem bei Kindern beliebten Gedächtnisspiel "Memory", hat jedoch noch einige zusätzliche Tücken.
Für das Spiel ist der Befehl STO M nötig. Es wird hier gezeigt, wie sich mit Hilfe des M-Registers das Alpharegister manipulieren läßt.

Spielanleitung:
Zu Spielbeginn werden vom 41C in 2o "Fächern" zehn Paare von Alphazeichen verteilt. Die erste Schwierigkeit gegenüber Memory ist schon ersichtlich, denn es gibt bei jedem Spiel neu zu merkende "Bilder", da mit STO M zufällige Alphabilder erzeugt werden.
Nun geben zwei Spieler abwechselnd zwei Fächernummern an (wiede eine Schwierigkeit: Man muß die Nummern beider Register angebem ehe man das erste sieht), und der 41C zeigt die Registerinhalte an (Achtung: Nur je zwei sec. zum Anschauen).
Sind die Inhalte von beiden Fächern gleich, so werden sie geleert und der Spieler erhält einen Punkt. Er darf <u>nicht</u> noch einmal raten.

Bedienungsanleitung:

1) Zufällige Zahlenfolge eingeben
2) STO oo
3) XEQ "MEM"
4) ca. 1 min. warten, dann: Anzeige: BITTE RATEN
   dann: Anzeige: NUMMER 1 U. 2?
5) Eingabe: 1. Nummer
6) Enter
7) Eingabe: 2. Nummer
8) Anzeige der Registerinhalte (je 2 sec.)
9) Wenn gleich: Anzeige: IHRE NUMMER?
   Eingabe: Spielernummer (1 oder 2)
   Wenn falsch: Weiter bei Schritt 4)
1o) Bei Spielende: Anzeige: ENDE
   dann: Anzeige: Punkte Spieler Nr. 1
   dann: Anzeige: Punkte Spieler Nr. 2
   dann: Anzeige: Spielernummer des Siegers
   END

Das Programm zeigt auf Knopfdruck den derzeitigen Anzeigemodus an. (z.B. FIX ∅ oder ENG 8) Für dieses Prgr. wird kein Speicherregister benutzt, das Alpharegister zerstört und der Stack verändert.
Das Programm wird mit XEQ "ST?" gestartet. Für weitere Informationen siehe Benutzerhandbuch, Seite 231 !

```
LBL"ST?"
" SCI"
FS? 4o
" FIX"
FS? 41
" ENG"
"⊢ "
o
FC? 39
GTO oo
1
LBL oo
FC? 38
GTO o1
2
+
LBL o1
FC? 37
GTO o2
4
+
LBL o2
FC? 36
GTO A
8
+
LBL A
ARCL X
ASTO X
END
```

Achtung: Es ist wichtig, daß die angezeigte Ziffer auf Position 6 im Alpharegister zu finden ist. Dies geschieht durch setzen eines Blanks (Space) vor das Wort und eines zwischen Wort und Ziffer. Dann wird das A-Register durch ASTO X CLA ARCL X auf 6 Stellen gekürzt. So werden etwaige Kommata oder Dezimalstellen aus der Anzeige geschnitten.

EDMUND WEITZ

Rechnerorganisation des HP-41C - Teil 2

Ausgehend vom gleichnamigen Artikel in der letzten Ausgabe möchte ich hier weiter über "unseren" 41C berichten:

Zunächst zur Stromversorgung:

Standardmäßig wird der Rechner durch vier Lady-Zellen am Leben erhalten. Diese Batterien haben in der Regel eine Kapazität von 5oo mAh und kosten (Alkali-Mangan) ca. 2,5o DM das Stück. Besonders bei Kartenleserbetrieb empfiehlt sich daher die Verwendung anderer Energiequellen, wobei der externe Anschluß von Baby-Batterien mit 5ooo mAh in einem Batteriehalter die günstigste Möglichkeit darstellt. Der Stecker muß leider selbst gebastelt werden. Wer sich nicht traut, kann von mir noch einen Batteriehalter mit Anschlußklipp und Stecker für 5,-- DM erhalten; ich habe noch zwei (!) übrig. Dabei ist man aber auf stationären Betrieb eingeschränkt. Empfehlenswerter ist natürlich schon der HP-Akku Satz, den Clubmitglieder bei Günter Lelarge 1o % billiger haben können. Doch: Man muß einen Drucker haben und über 1oo,--DM investieren. Dafür ist man dann aber auch alle "Stromversorgungssorgen" los.

Doch zurück zum Rechner:

Er arbeitet mit Spannungen von 4 bis 7 Volt =. Die Rechengeschwindigkeit ist kaum abhängig von der Versorgungsspannung, eher schon von den angeschlossenen Peripherieeinheiten: Je mehr, desto langsamer. Dies ist bei Uhr-, Timer-, etc. Programmen zu berücksichtigen. Nackt, also ohne Peripherieeinheiten, zieht der Rechner nur wenige uA (1uA = $1o^{-6}$ A). Dieser extrem niedrige Stromverbrauch (man bedenke: alle Daten werden erhalten!) ist dadurch möglich, daß der Rechner in den sogenannten SLEEP-Mode abrutscht; alle bei "AUS" nicht nötigen Stromkreise, wie z.B. CPU, Tastatur excl. ON/OFF Schalter, Anzeige u.s.w. werden abgeschaltet. Erst durch betätigen des ON Schalters wird die Tastatur, die CPU und die Anzeige aktiviert. Der Stromverbrauch ist immer noch sehr gering. Wenn aber eine Taste gedrückt wird, eine Berechnung oder ein Programm läuft, dann schnellt der Verbrauch bei voll belegten I/O Ports u.U. bis auf max. 2o mA hoch. Doch auch hier ist das "hochschnellen" ein falscher Ausdruck: Man denke an andere LED-Rechner, deren Anzeige allein ein Vielfaches von 2o mA verbraucht.

Für die Ansteuerung der LCD-Anzeige stehen nun drei Versorgungsspannungen zur Verfügung. Um den Kontrast der Anzeige zu wahren, driften diese drei Spannungen (1,1 2,2 & 3,3 V=) um -2o mV/°C.

Die Betriebsspannung wird ständig über einen Unterspannungs-Kontrollkreis kontrolliert und notfalls wird der BAT-Indikator aktiviert. Wurde BAT aktiviert, so weiß die CPU, daß keine weiteren Schaltkreise aktiviert werden können; z.B. nimmt der Kartenleser keine Karte mehr an, wenn der BAT Indikator aktiviert ist (das BAT Flag gesetzt ist), auch wenn die Versorgungsspannung inzwischen stabil und ausreichend hoch ist.

Im Vergleich zu diesen Verbrauchswerten enorm hoch, im Vergleich mit anderen Geräten enorm niedrig ist der Durst des Kartenlesers nach elektrischer Energie. Er verbraucht um 15o mA beim Kartenlesen und beim Kartenschreiben weit darüber. Was viele glauben, erweist sich aber als falsch: Der Kartenleser zieht nur sehr wenig (ca. 1uA) Strom, wenn der Motor nicht gerade läuft. Noch geringer ist der Verbrauch der MEMORY-Moduln und der ROM's. Fazit: Es ist völlig unrentabel und daher in keiner Weise zu empfehlen, die angeschlossenen Einheiten bei jeder etwas größeren Pause zu entfernen um Energie zu sparen. Man denke nur einmal an die Abnutzung der Goldkontakte. Bei sehr langen Rechenpausen, z.B. Urlaub o.ä., sollte man dann lieber die Batterien ganz entfernen. Dies ist bei nicht auslaufsicheren Zellen sowieso dringend ratsam. Noch ein Tip: Wird der Kartenleser angeschlossen, so fließt bis zum ersten Einschalten des Rechners ein merklicher Strom in den Leser: Nach dem Anschluß des Lesers immer erst einmal den Rechner einschalten !! Ich nehme an, daß die CPU die Anwesenheit des Lesers erst verarbeiten muß, dazu im SLEEP-Mode aber nicht in der Lage ist.

Damit bin ich schon bei der CPU, die von HP extra für diesen Rechner entwickelt wurde. Daten werden als 8, 14 oder 16 bit-Worte verarbeitet. Die CPU verfügt über 5 Arbeitsspeicher, die von HP mit A,B,C,M,N bezeichnet werden (aber nichts mit den von uns entdeckten gleichnamigen Registern zu tun haben), einem 8-bit Register, einem 14bit Status-Register, zwei Programmzeigern und vier Unterprogramm-Rücksprungadressen.

Die ROM-Adressen sind 16bit-Worte, die RAM-Adressen dagegen nur 1obit-Worte. Die ROM-Adressen lassen sich unterteilen: Die unteren 12 Bits sind die eigentliche Adresse, während die oberen vier Bits zur Auswahl eines von 16 möglichen ROM-Kreisen dienen. Von diesen 16 Adressen sind unteren drei "fest vergeben" für die 3 eingebauten System-ROM's. Die oberen acht Adressen sind für Einsteckmoduln (Prgr. Pac's) reserviert, woraus sich auch errechnen läßt, wieviel Prgr. Pacs sich anschließen lassen: Ein ROM besitzt 4kByte. Drei 12 k ROM's sind also schon zuviel; vier 8k ROM's sind möglich und ebenso zwei 12 k und ein 8k ROM. Die meisten HP-Prgr.-Pacs haben aber nur 4kByte, womit der Rechner (bei 4 I/O Ports) gar nicht voll ausgelastet werden kann.

Doch was ist nun mit den mittleren 5 ROM Adressen ??

Nun: Dies sind die Adressen der Peripherieeinheiten. Sollten sie auch variabel sein oder sind nur fünf Peripherieeinheiten möglich ? Ich bin mir nicht sicher, auch wenn ich gehört habe, daß HP genau 5 Peripherieeinheiten herausbringt (bringen will), als da sind: Kartenleser, Drucker, Bar-Code-Leser, Kassetteninterface und ein sogenanntes Power Pac, das den Anschluß von dreizig bis vierzig Modulen ermöglichen soll

OLIVER RIETSCHEL

Referenz: Hewlett-Packard Journal ; March 198o

OR-ST-1

Für alle Freunde des synthetischen Programmierens: Ich habe inzwischen eine eigene Statuskarte mit folgenden Befehlen erstellt: RCL, STO, X   , RCL IND, STO IND, X   IND jeweils für die Register a, b, c, d, e, M, N, O, P, Q, ⊢. Wer Interesse hat, der möge mir zwei Magnetkarten schicken. Achtung: Die erste Kartenseite wurde nicht aufgezeichnet, so daß Anzeige, Stack, Alpharegister, Size,... nicht verändert werden. Nach dem Einlesen der drei Seiten ist daher Clear zu drücken.

OLIVER RIETSCHEL

```
USER KEYS:
 11 XROM 01,59
-11 XROM 05,59
 12 XROM 01,60
-12 XROM 05,60
 13 XROM 01,61
-13 XROM 05,61
 14 XROM 01,62
-14 XROM 05,62
 15 XROM 01,63
-15 XROM 05,63
 21 XROM 01,53
-21 XROM 05,53
 22 XROM 01,54
-22 XROM 05,54
 23 XROM 01,55
-23 XROM 05,55
 24 XROM 01,56
-24 XROM 05,56
 25 XROM 01,57
-25 XROM 05,57
 32 XROM 01,58
-32 XROM 05,58
 33 XROM 03,59
-33 XROM 07,59
 34 XROM 03,60
-34 XROM 07,60
 35 XROM 03,61
-35 XROM 07,61
 41 XROM 03,62
 42 XROM 03,63
-42 XROM 07,63
 43 XROM 03,53
-43 XROM 07,53
 44 XROM 03,54
-44 XROM 07,54
 51 XROM 03,55
 52 XROM 03,56
-52 XROM 07,56
 53 XROM 03,57
-53 XROM 07,57
 54 XROM 03,58
-54 XROM 07,58
 61 XROM 57,59
-61 XROM 59,59
 62 XROM 57,60
-62 XROM 59,60
 63 XROM 57,61
-63 XROM 59,61
 64 XROM 57,62
-64 XROM 59,62
 71 XROM 57,63
-71 XROM 59,63
 72 XROM 57,53
-72 XROM 59,53
 73 XROM 57,54
-73 XROM 59,54
 74 XROM 57,55
-74 XROM 59,55
 81 XROM 57,56
-81 XROM 59,56
 82 XROM 57,57
-82 XROM 59,57
 83 XROM 57,58
-83 XROM 59,58
```

KALENDER
AUSDRUCK

PROGRAMM NR. 10010 DER HP-USER-BIBLIOTHEK DIENT ZUM AUSDRUCK EINES KALENDERS MIT DEM DRUCKER DES HP-41C.
ES LAESST SICH JEDOCH UM 73 (!!) REGISTER VERKUERZEN, WIE IM FOLGENDEN GESCHEHEN.
ES IST DAMIT AUCH ENORM KUERZER ALS DAS IM FUNKSCHAU-SONDERHEFT ERSHIENENE PROGRAMM.

SIZE: 006
START: KAL
EINGABE: MONAT.JAHR
UND DIE ANZAHL ZU
DRUCKENDER MONATE

```
 01♦LBL "KAL"
 02 CF 29
 03 FIX 0
 04 "MM.JJJJ?"
 05 PROMPT
 06 INT
 07 LASTX
 08 FRC
 09 1 E4
 10 *
 11 STO 00
 12 RDN
 13 ENTER↑
 14 ENTER↑
 15 "MONATE?"
 16 PROMPT
 17 +
 18 1
 19 -
 20 ,001
 21 *
 22 +
 23 STO 01
 24♦LBL 15
 25 RCL 01
 26 INT
 27 STO 02
 28 RCL 00
 29 STO 03
 30 XEQ 19
 31 STO 05
 32 RCL 03
 33 STO 04
 34 1
 35 ST+ 02
 36 RCL 00
 37 STO 03
 38 XEQ 19
 39 RCL 03
 40 RCL 04
 41 -
 42 STO 04
 43 SF 12
 44 XEQ IND 01
 45 RCL 00
 46 ACX
 47 29
 48 SKPCOL
 49 ADV
 50 "S"
 51 ACA
 52 4
 53 SKPCOL
 54 "M"
 55 ACA
 56 SKPCOL
 57 "D"
 58 ACA
 59 SKPCOL
 60 "M"
 61 ACA
 62 SKPCOL
 63 "D"
 64 ACA
 65 SKPCOL
 66 "F"
 67 ACA
 68 SKPCOL
 69 "S"
 70 ACA
 71 6
 72 SKPCOL
 73 ADV
 74 CF 12
 75 XEQ 17
 76 ADV
 77 ADV
 78 RCL 01
 79 INT
 80 12
 81 X<=Y?
 82 XEQ 13
 83 ISG 01
 84 GTO 15
 85 RTN
 86♦LBL 13
 87 1
 88 ST+ 00
 89 12
 90 ST- 01
 91 ,001
 92 *
 93 ST- 01
 94 RTN
 95♦LBL 17
 96 RCL 04
 97 1 E3
 98 /
 99 RCL 05
100 1
101 -
102 CHS
103 X<0?
104 SF 05
105 ABS
106 +
107 FS?C 05
108 CHS
109 STO 05
110 0
111 ENTER↑
112 124
113 BLDSPEC
114 STO 02
115♦LBL 14
116 1,007
117 STO 03
118 CLA
119 ARCL 02
120♦LBL 18
121 ISG 05
122 SF 05
123 RCL 05
124 INT
125 X<=0?
126 CF 05
127 FC? 05
128 GTO 00
129 10
130 X>Y?
131 "⊦ "
132 RDN
133 ARCL X
134♦LBL 00
135 FC?C 05
136 "⊦  "
137 ARCL 02
138 ISG 03
139 GTO 18
140 "⊦ "
141 ACA
142 ADV
143 RCL 05
144 INT
145 RCL 05
146 FRC
147 1 E3
148 *
149 X<=Y?
150 RTN
151 GTO 14
152♦LBL 19
153 RCL 03
154 RCL 02
155 1
156 +
157 ENTER↑
158 1/X
159 ,7
160 +
161 CHS
162 INT
163 ST+ 03
164 12
165 *
166 -
167 30,6001
168 *
169 INT
170 RCL 03
171 365,25
172 *
173 INT
174 +
175 STO 03
176 7
177 MOD
178 X=0?
179 7
180 RTN
181♦LBL 01
182 "JANUAR"
183 18
184 GTO 00
185♦LBL 02
186 "FEBRUAR"
187 14
188 GTO 00
189♦LBL 03
190 "MAERZ"
191 25
192 GTO 00
193♦LBL 04
194 "APRIL"
195 25
196 GTO 00
197♦LBL 05
198 "MAI"
199 32
200 GTO 00
201♦LBL 06
202 "JUNI"
203 28
204 GTO 00
205♦LBL 07
206 "JULI"
207 28
208 GTO 00
209♦LBL 08
210 "AUGUST"
211 21
212 GTO 00
213♦LBL 09
214 "SEPTEM"
215 11
216 GTO 01
217♦LBL 10
218 "OKTO"
219 18
220 GTO 01
221♦LBL 11
222 "NOVEM"
223 14
224 GTO 01
225♦LBL 12
226 "DEZEM"
227 14
228♦LBL 01
229 "⊦BER"
230♦LBL 00
231 ACA
232 SKPCOL
233 ADV
234 END
```

BEISPIEL:

```
   SEPTEMBER
     1980
 S M D M D F S
|  | 1| 2| 3| 4| 5| 6|
| 7| 8| 9|10|11|12|13|
|14|15|16|17|18|19|20|
|21|22|23|24|25|26|27|
|28|29|30|  |  |  |  |

    OKTOBER
     1980
 S M D M D F S
|  |  |  | 1| 2| 3| 4|
| 5| 6| 7| 8| 9|10|11|
|12|13|14|15|16|17|18|
|19|20|21|22|23|24|25|
|26|27|28|29|30|31|  |
```

```
                CAT 1
LBL'KAL
.END.      455 BYTES
```

JIM DE ARRAS

HIGH RESOLUTION PLOTTING

DIESES PRGR. ZEICHNET
DEN GRAPHEN EINER PROGR.
FUNKTION.
DER UNTERSCHIED ZU
"PRPLOT" BESTEHT IN
DER DEUTLICH HOEHEREN
AUFLOESUNG.
LEIDER BENOETIGT MAN ZUR
AUSFUEHRUNG ENORM VIEL
GEDULD. ES GEHT IM
SCHNECKENTEMPO VORAN.
ES HANDELT SICH UM EINE
VERBESSERTE VERSION EI-
NES PRGR. VON J SCHWARTZ
AUS PPC J. JANUAR 1980

OLIVER RIETSCHEL

SIZE: 026
EIN MEMORY-MODUL
START MIT: XEQ PLOTTER
RESTART MIT: NEW

```
 01◆LBL "PLO
TTER"
 02 AON
 03 "NAME?"
 04 PROMPT
 05 AOFF
 06 ASTO 05
 07 "Y MIN?"
 08 PROMPT
 09 STO 00
 10 "Y MAX?"
 11 PROMPT
 12 STO 01
 13 "X MIN?"
 14 PROMPT
 15 STO 02
 16 "X MAX?"
 17 PROMPT
 18 STO 03
 19◆LBL "NEW
"
 20 "X INC?"
 21 PROMPT
 22 STO 04
 23 "PLOT OF
"
 24 ARCL 05
 25 PRA
 26 FIX 1
 27 "Y: "
 28 ARCL 00
 29 "├ TO "
 30 ARCL 01
 31 PRA
 32 "X: "
 33 ARCL 02
 34 "├ TO "
 35 ARCL 03
 36 PRA
 37 FIX 3
 38 "STEPS O
F "
 39 ARCL 04
 40 PRA
 41 "-------"
 42 ASTO L
 43 ARCL L
 44 ARCL L
 45 ARCL L
 46 PRA
 47 167
 48 RCL 01
 49 RCL 00
 50 -
 51 /
 52 STO 08
 53 RCL 02
 54 STO 06
 55◆LBL 00
 56 11,017
 57 STO 07
 58◆LBL 01
 59 RCL 06
 60 XEQ IND
05
 61 RCL 00
 62 -
 63 RCL 08
 64 *
 65 1
 66 +
 67 FIX 0
 68 RND
 69 168
 70 X<>Y
 71 X>Y?
 72 0
 73 ENTER↑
 74 0
 75 X<=Y?
 76 X<>Y
 77 STO IND
07
 78 RCL 04
 79 ST+ 06
 80 RCL 06
 81 RCL 03
 82 X<Y?
 83 GTO 04
 84 ISG 07
 85 GTO 01
 86 XEQ 05
 87 4
 88 RCL 04
 89 *
 90 ST+ 06
 91 RCL 06
 92 RCL 03
 93 X<Y?
 94 RTN
 95 GTO 00
 96◆LBL 04
 97 0
 98 STO IND
07
 99 ISG 07
100 GTO 04
101◆LBL 05
102 ,001
103 ST+ 11
104 ,002
105 ST+ 12
106 ,004
107 ST+ 13
108 ,008
109 ST+ 14
110 ,016
111 ST+ 15
112 ,032
113 ST+ 16
114 ,064
115 ST+ 17
116 12,017
117 STO 10
118◆LBL 06
119 RCL 10
120 STO 09
121 1
122 -
123 RCL IND
X
124◆LBL 07
125 RCL IND
09
126 X<Y?
127 X<>Y
128 STO IND
09
129 RDN
130 ISG 09
131 GTO 07
132 STO IND
Y
133 ISG 10
134 GTO 06
135 11,016
136 STO 09
137 11,017
138 STO 10
139◆LBL 08
140 ISG 10
141 RCL IND
10
142 INT
143 RCL IND
09
144 INT
145 X≠Y?
146 GTO 09
147 LASTX
148 FRC
149 ST+ IND
10
150 CLX
151 STO IND
09
152◆LBL 09
153 ISG 09
154 GTO 08
155 11,017
156 STO 09
157 0
158 STO 10
159◆LBL 10
160 RCL IND
09
161 INT
162 X=0?
163 GTO 11
164 ENTER↑
165 X<> 10
166 -
167 1
168 -
169 SKPCOL
170 RCL IND
09
171 FRC
172 1 E3
173 *
174 ACCOL
175◆LBL 11
176 ISG 09
177 GTO 10
178 PRBUF
179 RTN
180 END
```

```
                    CAT 1
LBL'PLOTTER
LBL'NEW
END          388 BYTES
LBL'SPEZI
END           14 BYTES
.END.         04 BYTES
```

BEISPIEL:

```
PLOT OF SPEZI
Y: -1.0 TO 1.0
X: 0.0 TO 360.0
STEPS OF 1.000
```

ENDE DES BEISPIELS
DRUCKDAUER: CA. 10 MIN.

HIERBEI WURDE BENUTZT:

```
 01◆LBL "SPEZI"
 02 SIN
 03 X↑2
 04 END
```

ZUM VERGLEICH NOCH EIN-
MAL MIT "PRPLOT" :

```
      PLOT OF SPEZI
    X (UNITS= 1.) ↓
    Y (UNITS= 1.) →
     -1.00          1.00
            0.00
    ├────────┼────────┤
  0.         *
 20.         :*
 40.         :   *
 60.         :      *
 80.         :        *
100.         :        *
120.         :      *
140.         :   *
160.         :*
180.         *
200.         :*
220.         :   *
240.         :      *
260.         :        *
280.         :        *
300.         :      *
320.         :   *
340.         :*
360.         *
```

DRUCKDAUER: CA. 30 SEC.

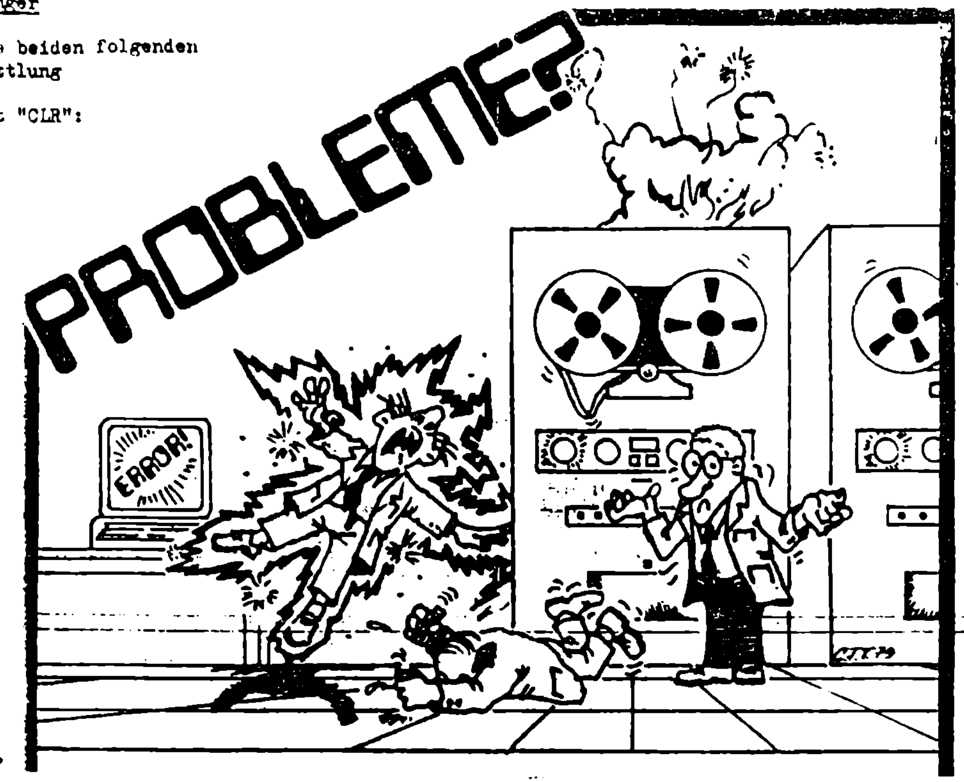


Standardwerke ? - Tips für Anfänger

Von Siegfried Schwarze stammen die beiden folgenden Routinen zur Speicherbereichsermittlung und Gesamtlöschung.
Zunächst zur Löschroutine, benannt "CLR":

CLR-ROUTINE
VON SIEGFRIED SCHWARZE

```
 01♦LBL "CLR
-
 02 CLRG
 03 CLST
 04 0
 05 STO L
 06 CF 00
 07 CF 01
 08 CF 02
 09 CF 03
 10 CF 04
 11 RTN
 12 END
```



Aus: Markt und Technik Juli 198o

Siegfried sah das RTN in Zeile 11 vor, damit ein Rücksprung nach einem Aufruf durch ein anderes Programm erfolgen kann.
Das ist jedoch nicht nötig, da END dieselbe Wirkung hat.
Statt der Befehle CF o1, CF o2,... empfehle ich den Befehl STO d einzufügen, so daß alle Flags gelöscht werden. Das kann vorteilhaft sein, muß aber nicht.
Damit ein gewohntes Anzeigebild entsteht, fügte ich noch einen FIX Befehl ein. Er kann auch weggelassen werden oder gegen jeden anderen FIX, ENG, SCI Befehl ausgetauscht werden (oder auch: 7DSP n) .

CLR-ROUTINE
VON OLIVER RIETSCHEL

```
 01♦LBL "CLR
-
 02 CLRG
 03 CLST
 04 0
 05 STO L
 06 STO d
 07 FIX 4
 08 END
```

Weiterhin schrieb Siegfried die folgende Routine zur Ermittlung der Speicherbereichsverteilung (SIZE):

SIZE-ROUTINE
VON SIEGFRIED SCHWARZE

```
 01♦LBL "S?"
 02 0
 03 STO 00
 04♦LBL 01
 05 SF 25
 06 RCL IND
00
 07 FS? 25
 08 GTO 00
 09 "SIZE: "
 10 ARCL 00
 11 PROMPT
 12♦LBL 00
 13 1
 14 ST+ 00
 15 GTO 01
 16 END
```

Unschön ist, daß das Register oo verwendet, d.h. auch zerstört wird. Außerdem bewirkt diese Tatsache, daß SIZE oo nicht angezeigt wird, sondern ein NONEXISTENT bewirkt.
Daher habe ich das Programm optimiert. Es läßt sich auch als Unterprogramm verwenden. Wem eine Anzeige "SIZE=..." lieber ist, kann hinter Zeile 11 noch ergänzen:
"SIZE=" ARCL X AVIEW

SIZE-ROUTINE
VON OLIVER RIETSCHEL

```
 01♦LBL "S?"
 02 SF 25
 03 -1
 04 X<>Y
 05♦LBL 01
 06 X<>Y
 07 1
 08 +
 09 RCL IND
X
 10 FS? 25
 11 GTO 01
 12 END
```

Wichtig! Wichtig! Wichtig! Wichtig! Wichtig! Wichtig!

Liebe Clubmitglieder !!!!

Helft bitte alle mit, unser prisma "groß und stark" zu machen. Schreibt Artikel zu allem was interessant sein könnte, schickt Programme und Neuigkeiten und und und - eben alles was andere Clubmitglieder interessieren könnte. Nur so kann unser Club zu dem werden, was jeder Club einmal werden will!
Daß alles (zumindest fast alles) auch eine Veröffentlichung Wert ist, ist an dieser Ausgabe von prisma hoffentlich zu erkennen.
Also: Viel Spaß und "Happy Programming"

Programm: BRUCHRECHNUNG

C. Mirgel

PROGRAM LISTING

PROGRAMMABLAUFLISTING / LISTAGE DU PROGRAMME / LISTATO DI PROGRAMMA

Page 1 of 2 — Juni 1980

Line	Keystrokes	Comments	Line	Key pressed	Comments
	LBL "BRUCH"			STOP	
	FIX Ø			LBL Ø4	Nenner des Er-
	Ø			RCL ØØ	gebnisses an
	TONE 7			STO Ø2	Programmanfang
	TONE 4			GTO Ø1	übertragen
	"GANZE ZAHL 1?"	Bruch 1		LBL "+"	Addition
	PROMPT			XEQ "AS"	
	STO ØØ	ganzz.Anteil1		ST+ Ø1	
	"ZAEHLER 1 ?"			GTO Ø2	
	PROMPT			LBL "-"	Subtraktion
	STO Ø1	Zähler 1		XEQ "AS"	
	"NENNER 1 ?"			ST- Ø1	
	PROMPT			GTO Ø2	
	STO Ø2	Nenner 1		LBL "*"	Multiplikation
	LBL Ø1	gemischte Z.		RCL Ø4	
	RCL ØØ	in unechten		ST* Ø1	
	RCL Ø2	Bruch umwan-		RCL Ø2	
	*	deln		RCL Ø5	
	ST+ Ø1			*	
	Ø			STO Ø8	
	TONE 5			GTO Ø2	
	TONE 6			LBL "/"	Division
	"GANZE ZAHL2?"	Bruch 2		RCL Ø5	
	PROMPT			ST* Ø1	
	STO Ø3	ganzz.Anteil2		RCL Ø2	
	"ZAEHLER 2 ?"			RCL Ø4	
	PROMPT			*	
	STO Ø4	Zähler 2		STO Ø8	
	"NENNER 2 ?"			GTO Ø2	
	PROMPT			LBL "EB"	
	STO Ø5	Nenner 2		Ø	
	RCL Ø3	gemischte Z.		STO ØØ	
	*	in unechten		RCL Ø8	Ergebnis:Nenner
	ST+ Ø4	Bruch umwand.		RCL Ø1	Ergebnis:Zähler
	"+, -, *, / ?"	Rechenart		X<=Y?	echter Bruch ?
	TONE 8			RTN	
	TONE 9			FIX 4	
	PROMPT			RCL Ø1	ganzzahligen
	LBL Ø2			RCL Ø8	Anteil er-
	XEQ "EB"			/	mitteln
	XEQ "K"			INT	
	RCL Ø1			FIX Ø	
	X=0?	wenn Zähl.=0		STO Ø3	
	GTO Ø3	ganze Z. anz.		RCL Ø1	Zähler er-
	XEQ "ANZ"			RCL Ø8	mitteln
	LBL Ø3	nur ganze		MOD	
	" "	Zahl anzei-		STO Ø1	
	ARCL ØØ	gen		RTN	
	AVIEW			LBL "ANZ"	Ergebnis an-
	BEEP			" "	zeigen

Programm: BRUCHRECHNUNG

C. Mirgel

PROGRAM LISTING

PROGRAMMABLAUFLISTING / LISTAGE DU PROGRAMME / LISTATO DI PROGRAMMA

Page 2 — Juni 1980

Line	Keystrokes	Comments	Line	Key pressed	Comments
	ARCL Ø3	ganzz. Anteil		RCL Ø4	
	"⊢"			RTN	
	ARCL Ø1	Zähler		END, REG Ø1	
	"⊢/"				
	ARCL ØØ	Nenner			
	AVIEW				
	BEEP				
	STOP				
	GTO Ø4				
	LBL "GGT"	GGT ermitteln			
	LBL Ø5				
	MOD				
	LASTX				
	X<>Y				
	X≠Ø?				
	GTO Ø5				
	RDN				
	RTN				
	LBL "KGV"	kgV ermitteln			
	ST* Y				
	X<>Y				
	STO Ø7				
	X<>Y				
	ST/ Y				
	XEQ "GGT"				
	RCL Ø7				
	X<>Y				
	/				
	RTN				
	LBL "K"	Ergebnis kür-			
	RCL Ø1	zen			
	RCL Ø8				
	XEQ "GGT"				
	STO Ø9				
	ST/ Ø1				
	RCL Ø9				
	ST/ Ø8				
	RTN				
	LBL "AS"	gemeinsamer			
	RCL Ø2	Programmteil			
	RCL Ø5	für Add/Subtr.			
	XEQ "KGV"	Hauptnenner			
	STO Ø8	ermitteln			
	RCL Ø2	Bruch 1 auf			
	/	Hauptnenner			
	ST* Ø1	erweitern			
	RCL Ø8	Bruch 2 auf			
	RCL Ø5	Hauptnenner			
	/	erweitern			
	ST* Ø4				

Bemerkungen:

SIZE 010

Vorschlag für Tastenzuordnungen:

LBL "BRUCH"	Taste A
LBL "+"	Taste +
LBL "-"	Taste -
LBL "*"	Taste *
LBL "/"	Taste /

Bei echten Brüchen für "GANZE ZAHL ?" einfach Taste R/S drücken.

Soll das Ergebnis aus der Verknüpfung von Bruch 1 und 2 mit einem dritten Bruch verknüpft werden, so kann nach der Ergebnisanzeige Taste R/S gedrückt werden. Das Ergebnis wird dann als Bruch 1 gespeichert und der Rechner fragt gleich nach Bruch 2.

Ganze Zahlen werden zweckmäßigerweise als Bruch eingegeben: zum Beispiel: 5 als $\frac{5}{1}$.

Rechnerorganisation des HP-41C - Teil 3

Jedes Clubmitglied - sofern ein Kartenleser nicht fehlt - sollte inzwischen das Key-Assignment-Programm besitzen. Davon ausgehend hier die neuesten Informationen über die Hexcode-Tabelle und deren Anwendung im KA-Prgr.:
Auf den beiden folgenden Seiten ist die kombinierte Hexcode-Tabelle des HP-41C zu finden, die ich aus PPCJ V6 N5 P23-24 ergattern konnte und ergänzte.
Der Rechner ermöglicht 256 (16x16) verschiedene Hexcodes nämlich o bis 255 dezimal. Die 256 Kästchen sind mit diesen dezimalen Codes durchnummeriert. Dahinter ist jeweils das Zeichen zu finden, das der Drucker mit dem Befehl ACCHR bildet, der ASCII Code des Wertes. Darunter ist zu sehen, wie sich das jeweilige Zeichen in der Anzeige des Rechners darstellt (Bei vielen Nur-Drucker-Zeichen:)
Sind die jeweiligen Codes Programmcodes, wurden sie also im PRGM-Mode eingegeben, so haben sie die unter den Anzeigebildern stehende Bedeutung. Alle Zwei-Byte-Befehle benötigen - logischer Weise - noch einen zweiten Code, wie z.B. RCL nicht für sich alleine stehen kann. Es benötigt z.B. die 1o als zweites Byte, was RCL 1o ergibt. Diese zweiten Bytes findet man in den Kästchen in der zweiten Zeile von unten. Hier setzt das Key-Assignment-Programm an: Die ROM-programmierte Systemsprache des 41C läßt hier nur die Adressen oo bis 99 zu. Außerdem IND oo bis 99, X,Y,Z,T,L und IND X,....L . Übrig bleiben also noch 46 Lücken. Wie unter Rechnerorganisation Teil 1 gesagt, gleichen sich alle Zeilen in der Tabelle in der Ausführung, um eine Vereinfachung der Berechnung zu ermöglichen. Naheliegend also, daß in Zeile 6 nach 99 einfach die Adressen 1oo,1o1,1o2,...111 folgen. Und so ist es auch ! Mit Hilfe des KA-Prgr. läßt sich nun jeder beliebige 2-Byte Befehl (z.B. VIEW, RCL, STO, ST+,...) mit diesen Adressen verbinden. Sie geben auf die Abfrage "KEY?" einfach das erste Byte dezimal an (z.B. 144 für RCL), ENTER, dann das zweite Byte, die Adresse (z.B. 1o5 für D), ENTER, XY die Taste mit X-Zeile, Y-Spalte (siehe KA).
Doch - Oh Wunder ! - sind nicht nur die Adressen 1oo bis 111 möglich, sondern alle unsere lieben Register finden wir in Zeile 7 wieder ! Für RCL c gibt man einfach an: 144, ENTER, 125, ENTER, XY, R/S
Die Adressen laufen also von oo bis 111 und von T bis e fort. Ab Zeile 8 wiederholt sich aber das ganze - wieso ?? Nun- ganz einfach- hier sind die IND Adressen zu finden. RCL IND B wäre also folgendermaßen zu erzeugen:
144, ENTER, 231, ENTER, XY, R/S
Damit wäre das Geheimnis um KA zumindest zum Teil gelüftet. Zu der Hexcode-Tabelle läßt sich noch viel, viel sagen. Doch mehr im nächsten prisma und viel Spaß mit den neuen Möglichkeiten; nur noch soviel:
Vorsicht bei den Registern 1oo und 1o1; die Befehle sehen aus, wie für die Register oo und o1.
Die in den Kästchen ganz unten aufgeführten Befehle zeigen, wie der HP-67/97 die Codes interpretiert. Die Rätsel um die TEXT-Befehle werde ich im nächsten Artikel Rechnerorganisation lüften. Bis dahin erst einmal genug technischen Krams und noch ein Wort an alle "Nicht-Kartenleser-Besitzer" und/oder "Nicht-Drucker-Besitzer": Alle Kartenleser und Drucker-Befehle will ich in einem Artikel erklären. Für diese Ausgabe habe ich es aber nicht mehr geschafft. Daher im nächsten prisma mit frischer Energie und - hoffentlich - wieder ein paar Neuigkeiten,

OLIVER RIETSCHEL

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0 -- NULL 00 *R/S*	1 × LBL 00 01 *1/x*	2 x̄ LBL 01 02 *x^2*	3 ← LBL 02 03 *$\sqrt{x}$*	4 α LBL 03 04 *%*	5 β LBL 04 05 *Σ+*	6 Γ LBL 05 06 *y^x*	7 ↓ LBL 06 07 *ln*	8 Δ LBL 07 08 *e^x*	9 σ LBL 08 09 *R→P*	10 ◆ LBL 09 10 *SIN*	11 λ LBL 10 11 *COS*	12 μ LBL 11 12 *TAN*	13 ∡ LBL 12 13 *P→R*	14 τ LBL 13 14 *RTN*	15 Φ LBL 14 15 *RCL Σ+*
1	16 θ 0 16 *0*	17 Ω 1 17 *1*	18 δ 2 18 *2*	19 Å 3 19 *3*	20 á 4 20 *4*	21 Ä 5 21 *5*	22 ä 6 22 *6*	23 Ö 7 23 *7*	24 ö 8 24 *8*	25 Ü 9 25 *9*	26 ü . 26 *.*	27 Æ EEX 27 *ENTER↑*	28 æ CHS 28 *CHS*	29 ≠ GTOα 29 *EEX*	30 £ XEQα 30 *+*	31 ▒ SPARE 31 *(24)*
2	32 (space) RCL 00 32 *PAUSE*	33 ! RCL 01 33 *n!*	34 " RCL 02 34 *x̄*	35 # RCL 03 35 *s*	36 $ RCL 04 36 *%CH*	37 % RCL 05 37 *Σ-*	38 & RCL 06 38 *ABS*	39 ' RCL 07 39 *LOG*	40 (RCL 08 40 *10^x*	41) RCL 09 41 *INT*	42 * RCL 10 42 *$\sin^{-1}$*	43 + RCL 11 43 *$\cos^{-1}$*	44 , RCL 12 44 *$\tan^{-1}$*	45 - RCL 13 45 *FRAC*	46 . RCL 14 46 *RND*	47 / RCL 15 47 *(32 24)*
3	48 0 STO 00 48 *x<>y*	49 1 STO 01 49 *R↓*	50 2 STO 02 50 *CLX*	51 3 STO 03 51 *ENG*	52 4 STO 04 52 *FIX*	53 5 STO 05 53 *-x-*	54 6 STO 06 54 *SCI*	55 7 STO 07 55 *+*	56 8 STO 08 56 *-*	57 9 STO 09 57 *x*	58 : STO 10 58 *D→R*	59 ; STO 11 59 *R→D*	60 < STO 12 60 *H→HMS*	61 = STO 13 61 *HMS→H*	62 > STO 14 62 *STO (i)*	63 ? STO 15 63 *RCL (i)*
4	64 @ + 64 *HMS+*	65 A - 65 *SPACE*	66 B * 66 *STK*	67 C / 67 *LST x*	68 D x<y? 68 *W/DATA*	69 E x>y? 69 *MERGE*	70 F x≤y? 70 *X<>I*	71 G Σ+ 71 *R↑*	72 H Σ- 72 *π*	73 I HMS+ 73 *DEG*	74 J HMS- 74 *RAD*	75 K MOD 75 *GRD*	76 L % 76 *P<>S*	77 M %CH 77 *CL REG*	78 N P→R 78 *REG*	79 O R→P 79 *(35 73)*
5	80 P LN 80 *x≠y*	81 Q x^2 81 *x=y*	82 R $\sqrt{x}$ 82 *x>y*	83 S y^x 83 *x≠0*	84 T CHS 84 *x=0*	85 U e^x 85 *x>0*	86 V LOG 86 *x<0*	87 W 10^x 87 *x≤y*	88 X e^x-1 88 *F? 0*	89 Y SIN 89 *F? 1*	90 Z COS 90 *F? 2*	91 [TAN 91 *F? 3*	92 \\ ASIN 92 *ISZ*	93] ACOS 93 *ISZ(i)*	94 ↑ ATAN 94 *DSZ*	95 _ →DEC 95 *DSZ(i)*
6	96 ` 1/x 96 *DSP 0*	97 a ABS 97 *DSP 1*	98 b FACT 98 *DSP 2*	99 c x≠0? 99 *DSP 3*	100 d X>0? *DSP 4*	101 e LN1+x *DSP 5*	102 f x<0? A *DSP 6*	103 g x=0? B *DSP 7*	104 h INT C *DSP 8*	105 i FRC D *DSP 9*	106 j D→R E *CF 0*	107 k R→D F *CF 1*	108 l →HMS G *CF 2*	109 m →HR H *CF 3*	110 n RND I *(35 61 04)*	111 o →OCT J *DSP (i)*
7	112 p CLΣ T *RCL 0*	113 q x<>y Z *RCL 1*	114 r PI Y *RCL 2*	115 s CLST X *RCL 3*	116 t R↑ LAST x *RCL 4*	117 u RDN *RCL 5*	118 v LAST x *RCL 6*	119 w CLX *RCL 7*	120 x x=y? *RCL 8*	121 y x≠y? *RCL 9*	122 z SIGN *RCL A*	123 ϖ x≤0? a *RCL B*	124 ǀ MEAN b *RCL C*	125 → SDEV c *RCL D*	126 Σ AVIEW d *RCL E*	127 ⊦ CLD e *RCL I*

PPC JOURNAL

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
8	128 ◆ DEG 00 STO÷0	129 RAD 01 STO÷1	130 GRAD 02 STO÷2	131 + ENTER↑ 03 STO÷3	132 α STOP 04 STO÷4	133 RTN 05 STO÷5	134 BEEP 06 STO÷6	135 CLA 07 STO÷7	136 Δ ASHF 08 STO÷8	137 PSE 09 STO÷9	138 ◆ CLRG 10 SF 0	139 AOFF 11 SF 1	140 AON 12 SF 2	141 ∡ OFF 13 SF 3	142 PROMPT 14 (35 51 04)	143 ADV 15 STO÷(i)	One Byte
9	144 RCL 16 STO 0	145 STO 17 STO 1	146 STO+ 18 STO 2	147 STO- 19 STO 3	148 STO* 20 STO 4	149 STO/ 21 STO 5	150 ISG 22 STO 6	151 DSE 23 STO 7	152 VIEW 24 STO 8	153 Σ REG 25 STO 9	154 ASTO 26 STO A	155 ARCL 27 STO B	156 FIX 28 STO C	157 SCI 29 STO D	158 £ ENG 30 STO E	159 TONE 31 ST I	Two Byte Instructions
A	160 XROM 32 STO-0	161 ! XROM 33 STO-1	162 " XROM 34 STO-2	163 # XROM 35 STO-3	164 $ XROM 36 STO-4	165 % XROM 37 STO-5	166 & XROM 38 STO-6	167 XROM 39 STO-7	168 (SF 40 STO-8	169) CF 41 STO-9	170 * FS?C 42 GSB fa	171 + FC?C 43 GSB fb	172 , FS? 44 GSB fc	173 - FC? 45 GSB fd	174 . GTO IND XEQ IND 46 GSB fe	175 / SPARE 47 STO-(i)	
B	176 0 SPARE 48 GSB 0	177 1 GTO 00 49 GSB 1	178 2 GTO 01 50 GSB 2	179 3 GTO 02 51 GSB 3	180 4 GTO 03 52 GSB 4	181 5 GTO 04 53 GSB 5	182 6 GTO 05 54 GSB 6	183 7 GTO 06 55 GSB 7	184 8 GTO 07 56 GSB 8	185 9 GTO 08 57 GSB 9	186 GTO 09 58 GSB A	187 ; GTO 10 59 GSB B	188 < GTO 11 60 GSB C	189 = GTO 12 61 GSB D	190 > GTO 13 62 GSB E	191 ? GTO 14 63 GSB (i)	
C	192 @ αLBL 64 STO+0	193 A END 65 STO+1	194 B END 66 STO+2	195 C END 67 STO+3	196 D END 68 STO+4	197 E END 69 STO+5	198 F END 70 STO+6	199 G END 71 STO+7	200 H END 72 STO+8	201 I END 73 STO+9	202 J END 74 GTO fa	203 K END 75 GTO fb	204 L END 76 GTO fc	205 M END 77 GTO fd	206 N X<> 78 GTO fe	207 O LBL nn 79 STO+(i)	
D	208 P GTO 80 GTO 0	209 Q GTO 81 GTO 1	210 R GTO 82 GTO 2	211 S GTO 83 GTO 3	212 T GTO 84 GTO 4	213 U GTO 85 GTO 5	214 V GTO 86 GTO 6	215 W GTO 87 GTO 7	216 X GTO 88 GTO 8	217 Y GTO 89 GTO 9	218 Z GTO 90 GTO A	219 [GTO 91 GTO B	220 \ GTO 92 GTO C	221] GTO 93 GTO D	222 ↑ GTO 94 GTO E	223 _ GTO 95 GTO (i)	Three Byte
E	224 ` XEQ 96 STOX0	225 a XEQ 97 STOX1	226 b XEQ 98 STOX2	227 c XEQ 99 STOX3	228 d XEQ ØØ STOX4	229 e XEQ Ø1 STOX5	230 f XEQ A STOX6	231 g XEQ B STOX7	232 h XEQ C STOX8	233 i XEQ D STOX9	234 j XEQ E LBL fa	235 k XEQ F LBL fb	236 l XEQ G LBL fc	237 m XEQ H LBL fd	238 n XEQ I LBL fe	239 o XEQ J STOX(i)	
F	240 p T LBL 0	241 q TEXT 1 Z LBL 1	242 r TEXT 2 Y LBL 2	243 s TEXT 3 X LBL 3	244 t TEXT 4 LAST x LBL 4	245 u TEXT 5 M LBL 5	246 v TEXT 6 N LBL 6	247 w TEXT 7 O LBL 7	248 x TEXT 8 P LBL 8	249 y TEXT 9 Q LBL 9	250 z TEXT 10 R LBL A	251 π TEXT 11 a LBL B	252 ǀ TEXT 12 b LBL C	253 → TEXT 13 c LBL D	254 Σ TEXT 14 d LBL E	255 ⊢ TEXT 15 e (31 25 24)	Variable

Postfix Indirect

Hier ist es nun - das erste Heft unseres Clubs. Zusammen mit diesem Heft erfolgte eine ganze Reihe von Änderungen, doch mehr auf den "inneren" Seiten. Hier ein kleiner Wettbewerb:

W E R   M A L T   O D E R   Z E I C H N E T   E I N   K L E I N E S   O D E R

A U C H   G R Ö S S E R E S   T I T E L B I L D   F Ü R   U N S E R   H E F T ?

Das Original muß im Format DIN A4 sein und einen ausreichenden Platz für das Inhaltsverzeichnis bieten. Einfarbig und kontrastreich.

I N H A L T S V E R Z E I C H N I S

Wenn ich es als "Redaktionsschluß" bezeichnen darf: 13. September 198o
Mitgliederliste: 2o. September 198o
??????????????: 22. September 198o

Veraltete Seiten wurden ausgeschieden !!!

Einem Teil dieser Auflage liegt ein Prospekt der Firma Hewlett-Packard über Softwarezubehör bei. Dieser ist auch direkt bei HP, Vertriebsleitung, Frkft./M. erhältlich.

SIZE-TEST

..... Wenngleich inzwischen schon viele SIZE-Routinen veröffentlicht wurden, so möchte ich doch meine noch anbieten, obwohl sie vom Speicherbedarf mit 41 Bytes etwas dick geraten ist. Der Vorteil liegt aber eindeutig in den nahezu konstanten Laufzeiten, egal ob nun SIZE ooo oder SIZE 31o eingestellt sind (jeweils 3,5 bis 3,8 sec.) .
Das Diagramm stellt einen Vergleich mit den Laufzeiten Ihrer (Oliver R.) Routine aus prisma 9-8o dar. .....

Friedrich Hillebrandt

(Listings nächste Seite)

=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=0=

Auch das noch ! ?!!?

Wer meinen Rechnerorganisations-Bericht letzten Monat gelesen hat, wird es gemerkt haben: Ich war mit den Daten zum Stromverbrauch des Rechners ziemlich ungenau, den es lag etwas in der Luft:

Bei mehreren Verbrauchswerten, die mir inzwischen vorliegen, sind

D I F F E R E N Z E N   U M   E I N I G E

Z E H N T A U S E N D   P R O Z E N T   O F T

festzustellen. Nun ist es naheliegend, daß durch Verwendung verschiedener Meßgeräte besonders im unteren Meßbereich erhebliche Fehler verborgen sind. Daher habe ich zusammen mit einem Schulkameraden die "Freßsucht" des HP-41c überprüft. Ergebnis:

	Rechner 1	Funktion	Rechner 2	
I in mA	ca. 6	BEEP	1o,5	
	5,5	bel. Taste gedr.	1o,3	
	o,6	nur ein	4,6	
	o,oo5	aus	3,2	(keine Tippfehler)

Diese Messungen wurden mit dem gleichen - dem selben ! - Meßgerät durchgeführt. Dem besten, das wir in der Schule auftreiben konnten. Der Meßgerätefehler ist also NULL , er fällt beim Vergleich heraus.
Man beachte nur den letzten Wert - er ist am schockierendsten: Der eine Rechner zieht fast 7ooo (in Worten: Siebentausend) mal mehr Strom als der andere. Ich will hier nun keine Panikmache betreiben, aber ich möchte doch empfehlen, mal die Ströme zu messen; die meisten unter uns werden das wohl fertigbringen. Doch das ist nicht einmal nötig. In vielen Fällen sind die Unterschiede so gravierend, daß man es merken muß: Selbst bei geringem Kartenleserbetrieb halten die Batterien nicht viel länger, als 1 Monat. Ohne Kartenleser nicht länger als 4 Monate. Sollten der eine oder der andere dieses festgestellt haben so kann ich nur raten: Einschicken an HP. Zwar habe ich eine entsprechende Anfrage vor zwei Wochen an HP gerichtet, doch kam bis heute keine Antwort. Ich bin jedoch felsenfest davon überzeugt: Ein Garantiefall, selbst wenn die Garantiezeit schon abgelaufen sein sollte.
Mehr kann ich im Moment auch nicht dazu sagen - es war wohl auch schon genug schaurig - hoffentlich vernichtet HP nicht den einen oder anderen BUG bei der Reparatur

Oliver

sososososososososososososososososososososososososososososososososososososososososo

CLUBPROJEKT Nr. 3
Wird nächsten Monat genauer erläutert: Druckerpapier, schwarz-(und blau- (??)) druckend zu einem Spottpreis. Wieder eine Sammelbestellung.

8888888888888888888888888888888888888888888888888888888888888888888888888888888

CLUBPROJEKT Nr. 4
Veröffentlichung von Programmen in Lösungsbüchern: Entsprechende Verhandlungen laufen. Hoffentlich liegt nächsten Monat schon etwas konkretes vor.

ACXR

ACXR wird anstelle von ACX zum rechtsbuendigen Tabellenausdruck benutzt.
Bei Programmstart und -ende steht die zu druckende Zahl im X-Register, im Y-Register steht die maximale Anzahl der Vorkommastellen, ggf. vermehrt um fuehrende Leerstellen.

K. H. Hoenow 25.08.88

```
01+LBL "ACXR"
02 CF 29
03 RCL Y
04 2
05 -
06 RCL Y
07 RND
08 X=0?
09 GTO 01
10 ABS
11 LOG
12 INT
13 X<=0?
14 GTO 01
15 ST- Y
16+LBL 01
17 RDN
18 SKPCHR
19 RDN
20 RND
21 ACX
22 RTN
23 END
```

SIZE-ROUTINE

VON FRIEDR. HILLEBRANDT

```
01+LBL "S?"
02 CLST
03 511
04 FIX 0
05+LBL 00
06 INT
07 X=Y?
08 RTN
09 STO Z
10 RCL Y
11 SF 25
12+LBL 01
13 +
14 2
15 /
16 ARCL IND X
17 FC? 25
18 GTO 00
19 RND
20 STO Y
21 R↑
22 GTO 01
23 END
```

```
PLOT OF TM-S?
X (UNITS= 1) ↓
Y (UNITS= 1) →
   0,0          60,0
   3,0
  |---------------|
  0 :
 25 :
 50 :
 75 :
100 :
125 :
150 :
175 :
200 :
225 :
250 :
275 :
300 :
```

(Seite ein wenig "schmierig", da blaue Listings)

ERZEUGUNG VON HEX-ZEICHEN mit RCL d und STO

Walter Kropf

Befehl	Hex-Code	Flags: 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
RCL c	90 7D	1	0	0	1	0	0	0	0	0	1	1	1	1	1	0	1
STO c	91 7D	1	0	0	1	0	0	0	1	0	1	1	1	1	1	0	1
RCL R	90 7A	1	0	0	1	0	0	0	0	0	1	1	1	1	0	1	0
STO R	91 7A	1	0	0	1	0	0	0	1	0	1	1	1	1	0	1	0
RCL e	90 7F	1	0	0	1	0	0	0	0	0	1	1	1	1	1	1	1
STO e	91 7F	1	0	0	1	0	0	0	1	0	1	1	1	1	1	1	1
X<>N	CE 76	1	1	0	0	1	1	1	0	0	1	1	1	0	1	1	0

Flags entsprechend der Codetabelle setzen, RCL d, STO M, die beiden linken Alpha-Zeichen beinhalten den eingegebenen Hex-Code. Sie können leicht mit ASTO, ARCL, ASHF isoliert und zur späteren Verwendung mit dem Befehlsgewinnungsprogramm in Speicherregister abgelegt werden.

Das Verfahren mit RCL d ist nicht notwendig für HEX-Codes, für welche ein Zeichen im Alpha-Modus eingegeben werden kann. Hier genügt es, das Alpha-Zeichen einzugeben und das Programm zur Befehlsgewinnung zu starten.
Für HEX-Codes, welche nicht die Zeichen A-F enthalten, kann dieser auch wie folgt eingegeben werden: 1 EEX "Code", STO M, ASHF der Code steht dann wieder im Alpha-Register.
Zum Beispiel: für RCL d: 1 EEX 90, STO M, ASHF, Alpha, f APPEND, Σ, Alpha, Programm zur Befehlserzeugung starten.

ERZEUGUNG BELIEBIGER BEFEHLE NACH HEX'CODE TABELLE

Programm wie in Prisma 2/4-80 mit folgender Änderung:

```
203 STO 00
204 RCL M
205 STO IND 00
```

Programm auf Karte speichern. Nach meiner Erfahrung braucht der Rechner nicht gelöscht werden und muß das Programm nicht an erster Stelle im Speicher stehen. Allerdings hängt die Zeile, in welcher der gewünschte Befehl generiert wird, davon ab, wo das Programm im Speicher steht.
Hex-Code des Befehls laut obiger Anweisung im Alpha-Register speichern und Programm starten. Der Befehl kann durch löschen der überflüssigen Programmteile isoliert und auf Karte gespeichert werden.

Zu prisma 4/5-80:
Auflisten von Programmen.
Ich bin gerne bereit Programmlisten von Magnetkarten für Clubmitglieder auszudrucken. Bitte angeben in welcher Art das Listing gewünscht wird:"Trace-Mode"(wie Programm "Phasen, Finsternisse), NORM-Mode (Zeile für Zeile, bei jedem Label Zwischenraum) MAN-Mode(Zeile für Zeile, fortlaufend).
Als Vergütung schlage ich vor ÖS 4,- je Kartenspur plus Freiumschlag zur Rücksendung. Wenn Warenprobesäckchen und Dose von Kleinbildfilm mitgesandt wird, erfolgt Rücksendung des Druckstreifens ungefaltet. Bezahlung in Briefmarken! oder Int.Antwortsch.

Walter Kropf
Schönowitzstraße 14
A-8700 Leoben

Startrou[illegible] für Zufallezahlengenerator

Walter Kropf

```
01♦LBL a
02 PI
03 "R/S ON"
04 AVIEW
05 SF 11
06♦LBL 05
07 ST+ 00
08 FS? 11
09 GTO 05
10 "PROGR.FORTS."
```

Pseudozufallszahlengeneratoren, wie zum Beispiel im Programm 'MEMORY' verwendet, erfordern die Eingabe einer Startzahl zwischen 0 und 1. Hier ist die Möglichkeit der Manipulation durch Eingabe einer bekannten Zahl gegeben.
Das nebenstehende Programm vermeidet diese Umstand. Nach Programmstart wird in einer Endlosschleife die Zahl Pi aufaddiert.
Die Schleife wird verlassen durch kurzes Ausschalten des Rechners und wiedereinschalten. Das Programm wird dann automatis fortgesetzt.Die so gewonnene Startzahl steht in R 00. Da die Schleife in der Seku etwa 4mal durchlaufen wird, ist nach einig Sekunden der Stoppzeitpunkt nicht mehr zu bestimmen und die Startzahl unabhängig vom Spieler.

Lieber Herr Kropf !

Da Ihr Brief sicherlich auch für andere Clubmitglieder interessant ist, habe ich ihn heute mit abgedruckt. Ebenso Erzeugung von HEX-Zeichen , Leistungsmessung, Stoppuhr und Startroutine.
Ihre Programme Fouriertransformation und Inverse F. werden in einem der Weihnacht sonderhefte erscheinen. In dem Weihnachtssonderheft "Synthetische Programmierung" werden Sie (und alle anderen) viel,viel Interessantes auch in Bezug auf Ihre Frage finden.

Bleiben Sie weiter so aktiv !!

Oliver

PRIVATE ? - NEIN DANKE !

Wer hat es sich nicht schon mal überlegt - wie kann man PRIVATE - Programme "knacken". Es ist möglich, nur wie ? Hier will ich es nicht verraten, denn:
SOLLTE MAN ES ÜBERHAUPT VERÖFFENTLICHEN ???
Damit wäre dann die schöne PRIVATE - Funktion überflüssig, d.h. unwirksam .
Ein heikles Thema. Was meint Ihr dazu ???

Oliver

HARTE NUSS !

Eine kleine Aufgabe: Programmieren der INT-Funktion. An sich kein Problem, aber:
Verwenden Sie <u>nicht</u> INT oder FRC im Programm ! Was nun ? Man nehme RND und FIX 0 !
Falsch! Auch das geht nicht, denn der Anzeigezustand, z.B. ENG 7 soll erhalten bleiben. Also: Wie geht es ?
Ein Tip: Wer es nicht schafft, sollte sich "Rechnerorganisation Teil 1" noch einma genau durchlesen !

Auflösung im nächsten Heft

Oliver

Klaus Werner Hornow

2000 Hamburg/ [illegible]n 17.08.1980
Alaskawey [illegible]

Herrn
Oliver Rietschel
Hewlett Packard Anwender-Club
Postfach 373

2420 Eutin

Lieber Herr Rietschel!

Vielen Dank für "prisma" 9-80. Den Fragebogen erhalten Sie beiliegend zurück.

Sie haben mich auf das "Key-Assignment-Programm" neugierig gemacht; ich bitte Sie, es mir auf Karten zu kopieren, desgleichen das Programm "OR-ST-1". 4 Magnetkarten sind beigefügt.

Zum Programm "KAL" (23/9-80) einige Bemerkungen: Das optische Symmetrieempfinden wird belohnt, wenn in Zeile 183 21 statt 18 und in Zeile 187 18 statt 14 gesetzt wird. Da ohnehin 5 Kartenspuren benötigt werden, lassen sich noch folgende Ergänzungen einfügen: Seit 1975 fängt die Woche auch in Deutschland mit dem Montag an, daher

zwischen 29 u. 30: CF 06; 1975; X<=Y?; SF 06
zwischen 49 u. 50: 4; FS? 06; GTO 20
streichen: Z. 52
zwischen 53 u. 54: LBL 20
zwischen 70 u. 71: FC? 06; GTO 21; SKPCOL; "S"; ACA; LBL 21
zwischen 175 u. 176: FC? 06; GTO 22; 1; -; LBL 22

Nun zum Club: Es war vorauszusehen, daß der bisherige bescheidene Mitgliedsbeitrag bald nicht mehr ausreichen würde. Auch ich halte DM 50 im Normalfall für angemessen, allerdings mit jährlicher Zahlung, um Buchungsgebühren zu sparen, und mit Rechnung. Bei der steigenden Mitgliederzahl und den damit verbundenen höheren Geldumwälzungen wäre allerdings eine Eintragung ins Vereinsregister zu erwägen.

Mit freundlichen Grüßen

K. W. Hornow

Anlagen: Fragebogen, 4 leere Magnetkarten

Programmdokumentation

KÜRZEN VON BRÜCHEN ================== Andreas Weiler

Dieses Programm kürzt einen Bruch Z/N auf die einfachste Form.

Zuerst tippt man das Programm laut Auflistung ein. Danach Startet man es mit XEQ "BK". Jetzt wird der ungekürzte Zähler Z_u mit R/S eingegeben, darauf der ungek. Nenner N_u.

Drückt man nun auf die Run-Taste wird der gekürzte Bruch folgendermaßen angezeigt: Z_g / N_g

Listing

```
01♦LBL "BK"
02 FIX 0
03 STO 07
04 STOP
05 STO 08
06 STO 09
07♦LBL 08
08 RCL 09
09 -
10 X=0?
11 GTO 27
12 X>0?
13 GTO 08
14 RCL 09
15 +
16 X<> 09
17 GTO 08
18♦LBL 27
19 RCL 07
20 XEQ A
21 STO 07
22 RCL 08
23 XEQ A
24 STO 08
25 CLA
26 ARCL 07
27 "⊢ / "
28 ARCL 08
29 PROMPT
30 GTO "BK"
31♦LBL A
32 RCL 09
33 /
34 END
```

10 Register

Speicherbelegung:

Ø7: Z
Ø8: N
Ø9: ggT(Z,N)

ooooooooooooooooooooooooooooooooooooooooooooooooooooo

HP-User-Bibliothek Europa

Wie im prisma bereits angekündigt war: Der neue HP-41c Katalog ist erschienen. Zumindest quantitativ hat sich ja einiges getan; die negativen Befürchtungen wurden - zum Glück - nicht war. Wie es nun mit der Qualität steht, wird sich noch herausstellen müssen. Wer kann berichten ??

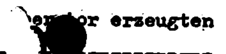
Oliver

Weil hier noch Platz war, hier - mitten im Heft - ein Hinwei[s] zur Mitgliederliste:
Bei Speichermodulen habe ich nur eine Unterscheidung getroff[en]
Wer nur eins besitzt oder bestellt hat, findet bei sich: Memory. Wer mehrere bestellt hat oder schon besitzt: Memorys

Bei einigen war für die Aufzählung der ROM's kein Platz mehr. Darum hier:

Mitgl. Nr.	ROMS
2	Games, Network Analysis
9	Games, Mathe, Standard
11	Mathe, Statistic, Standard, Circuit Analy[sis]

Oliver

Programm zur Überprüfung der durch einen Zufallszahlen**r erzeugten Zahlen**

```
01◆LBL "KDM"
02 CLRG
03 FIX 0
04 .258369
05 STO 50
06 50
07 STO 51
08 CF 29
09◆LBL 01
10 1
11 ST+ 53
12 RCL 53
13 VIEW X
14 XEQ "RNDM"
15 STO 52
16 1
17 ST+ IND 52
18 GTO 01
19◆LBL A
20 CLST
21 CLA
22◆LBL 02
23 "Z"
24 ARCL Y
25 "⊦ "
26 ARCL IND Y
27 AVIEW
28 PSE
29 RDN
30 1
31 ST+ Y
32 RDN
33 RCL 51
34 X>Y?
35 GTO 02
36 STOP
37 GTO 01
38◆LBL "RNDM"
39 RCL 50
40 9821
41 *
42 .211327
43 +
44 FRC
45 STO 50
[illegible]
47 RCL 51
48 *
49 INT
50 END
```

Das Programm speichert die Anzahl der generierten Zufallszahlen im Register mit der Nummer, die der Zufallszahl entspricht. Diese Register können dann durch XEQ A angezeigt werden. Während des Programmablaufs wird ständig die Zahl des bisher generierten Zahlen angezeigt.
Das Programm kann jederzeit durch R/S gestoppt, die Register mit XEQ A angesehen und mit R/S weitergefahren werden.

Status: Size= höchste zu generierende Zahl +5

Register n+1= Ausgangszahl für den Zufallsgenerator
" n+2= Höchstzahl +1
" n+3= Zufallszahl
" n+4= Anzahl der generierten Zahlen

n= Höchstzahl (im Beispielprogramm=49)

Erwin Hartmann

%&%

Zum Kalenderprogramm der Standardsammlung noch eine weitere Erweiterung: "Zinsprogramm"
Dieses Programm dient zur Kontrolle der Zinsrechnung von Sparkassenbüchern, insbesondere, wenn sich die Zinsen (Einlagen) im Jahre oft ändern. Man kommt natürlich nicht genau an die richtigen Werte heran, sonst muß man sich von den Banken die genaue Wertstellung sagen lassen und die Berechnung der Tage im Programm etwas ändern.

Heinrich Henze

```
01◆LBL E
02 RCL 10
03 *
04 360
05 /
06 STOP
07 *
08 ST+ 11
09 END
```

3+4+5+6+7+8+9+1+2+3+4+5+6+7+8+9+1+2+3+4+5+6+7+8+9+1+2+3

Mal etwas ganz, ganz anderes:

Verkaufe a) Graupner Fernsteuerungssender 8 S FM4o 8Kanal, 4o MHz, neuwertig, lediglich Antenne leicht geknickt, Akku VHB 15o DM

b) Equalizer Bausatz 4x1o Frequenzen, 2 Frontplatten, 2o Potis (Stereo), Klangregeleinheiten fertig aufgebaut für 2o Frequenzen Preis Verhandlungssache

Oliver

Additionsprogramm

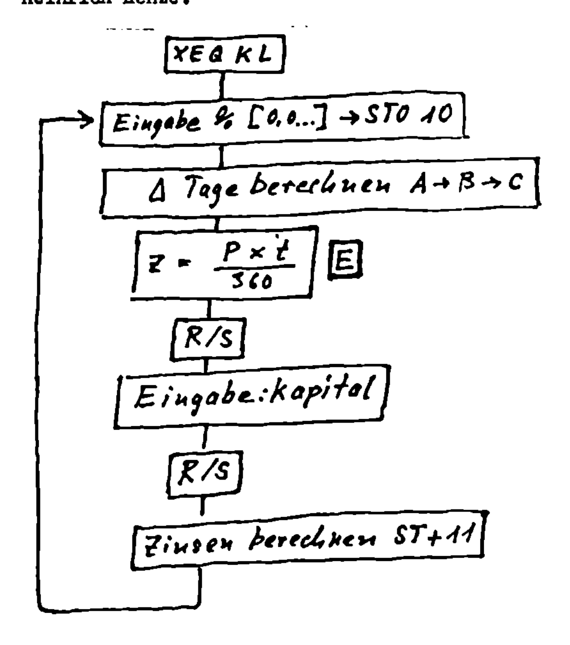
........Diesem Schreiben lege ich das Programm Addition bei. Dieses Programm dient zur Addition von Werten und anschließender Kontrolle, ob die Werte richtig eingegeben wurden, wenn kein Drucker vorhanden ist. Nach Starten des Programmes sind die Werte einzugeben, danach ist jeweils Taste R/S zu drücken. Wenn die Eingabe beendet ist, wird abermals die Taste R/S gedrückt. Die eingegebenen Werte werden dann automatisch wiedergegeben, sowie die Anzahl der Werte und die Summe.
Das Programm ist auf SIZE 4o ausgelegt. Durch Ändern der Programmzeile o3 in ,o14o1 kann das Programm z.B. auf SIZE 2o abgeändert werden. Bei der Eingabe des letzten Wertes vor Erreichen der Speichergrenze ertönt BEEP .
Durch drücken der R/S Taste kann dieser Wert gerettet werden. Durch nochmaliges Drücken der R/S Taste geht das Programm dann in eine Wiedergaberoutine.

Heinz Elsässer

```
01•L [illegible]
02 CL[illegible]
03 .03401
04 STO 00
05 4
06 STO 01
07 CLX
08 "ADDITIO
N"
09 PROMPT
10 STO IND
01
11 1
12 ST+ 01
13 RDN
14 STOP
15•LBL 02
16 ISG 00
17 GTO 03
18 BEEP
19 STOP
20•LBL 03
21 STO IND
01
22 +
23 1
24 ST+ 01
25 RCL Y
26 STOP
27 FC?C 22
28 GTO 01
29 GTO 02
30•LBL 01
31 STO 03
32 RCL 01
33 4
34 -
35 STO 02
36 RCL 04
37 PSE
38 5
39 STO 04
40 RCL IND
04
41 PSE
42•LBL 04
43 1
44 ST+ 04
45 RCL IND
04
46 PSE
47 X≠0?
48 GTO 04
49 RCL 02
50 CLA
51 ARCL X
52 "⊢ WERTE
"
53 AVIEW
54 PSE
55 RCL 03
56 "SUMME="
57 ARCL X
58 AVIEW
59 END
```

:=:

Hier noch das Flußdiagramm für das Zinsprogramm von Heinrich Henze:



Speicher: o1 Differenz in Tagen
1o Zinsen (Z.B. o,o4)
11 Zinskonto

:=:

Österreichische Mitglieder können Programme gegen Rückporto bei ROBERT KLAUC auflisten lassen.

Umrechnung von Dezimalsystem in jedes beliebige System kleiner 1o

```
01+LBL "ZSY"
02 FIX 0
03 "SYSTEM?"
04 PROMPT
05 STO 04
06 "ZAHL?"
07 PROMPT
08+LBL 01
09 0
10 STO 01
11 STO 03
12 RDN
13+LBL 00
14 STO 00
15 RCL 04
16 /
17 INT
18 STO 02
19 RCL 00
20 RCL 04
21 MOD
22 RCL 01
23 10↑X
24 *
25 ST+ 03
26 1
27 ST+ 01
28 RCL 02
29 X≠0?
30 GTO 00
31 VIEW 03
32 STOP
33 GTO 01
34 END
```

SIZE ØØ5 , Start mit XEQ'ZSY , weitere Zahlen , R/S ;
neues System natürlich mit RTN ; Ergebnis mit max 1Ø Stellen .
FIX Ø kann weggelassen werden und von Hand eingegeben werden . Setzen oder löschen von Flag 28 und 29 ist reine Geschmacksache .

- -

128 Töne auf dem HP - 41 C

Es besteht die Möglichkeit , mit jedem HP - 41 C einhundertachtund - zwanzig Töne zu erzeugen . Eine genaue Anleitung , wie diese Befehle zu erzeugen sind , ist noch nicht erstellt , befindet sich aber in Arbeit .
Fertige Statuskarten mit diesen Tönen sind gegen Einsendung von vier Magnetkarten und selbstadressierten Rückumschlag bei Walter Pieper - hoff (2o5) erhältlich .

- -

Freigehalten für allerletzte Neuigkeiten :

zu Seite 36 : Inzwischen bin ich der Meinung , wir sollten nicht verraten , wie PRIVATE geknackt wird .
Oder ? Oliver

REVERSI : von Ulrich Davertzhofen stammt ein Reversi spielendse Programm . Genaue Einzelheiten habe ich noch nicht abgeklärt .

ACHTUNG! ACHTUNG! ACHTUNG !! ACHTUNG! ACHTUNG! ACHTUNG? ACHTU NG ! !

Heute am 29.o9.8o ist ein Brief von HP angekommen : in Kürze die wichtigsten Auszüge :

Stromverbrauchsprobleme : Interne Verriegelung durch statische Entladung

..... Der Preis für den opt. Lesestift beträgt incl. MWST 297,- DM ..

Lieferzeiten : HP 41 C .... 2 Wochen
Kartenleser .... 4 Wochen
Drucker .... 5 Wochen

Zu diesen Lieferzeiten bekommt jeder Händler in Deutschland die Geräte !
Soweit der HP - Brief von Herrn Ulrich Brode , Innendienstingenieur bei Hewlett - Packard .
Ich möchte neueste Erfahrungswerte für Norddeutschland nachliefern :
Preise merklich nachgebend ! Häufig 1o bis 15 % unter Listenpreis .
Sehr oft ab Lager ! Aus diesen Gründen kann ich im Moment auch keine billigeren Händler empfehlen . Lohnend:Sammelbestellungen. Oliver

Hallo Oliv[illegible] **Hallo Ihr anderen [illegible]**

Deine gewün[illegible] Angaben zu meiner Person liegen diesem Brief bei, gleichzeitig sende ich Euch meine ersten Programme.

1. ON/OFF

Dieses Programm zeigt, daß auch eiskalt kalkulierende (Taschen)rechner fähig sind, menschlich zu reagieren.

LBL "ON/OFF"	
"GOOD BYE"	Off-Text
AVIEW	
SF 11	On-Vorbereitung
CF 27	UserMode Off
TONE 9	Abschiedssignal
OFF	Abschalten
"HALLO AXEL"	ON-Text
AVIEW	
END	

Das Programm verändert mit Ausnahme des Alpha-Registers keine Speicher.
ON/OFF wird einer dem Anwender genehmen Taste zugeordnet. Anstatt den Rechner mit ON abzuschal[ten] wird dann diese Taste benutzt.
Der Rechner wird ganz normal mit ON angeschalte[t]. Wurde nach dem Anschalten des Rechners kein Pro[gramm] ausgeführt, reicht es, den Rechner mit R/S abzuschalten.
Das "CF 27" (User Off) ist mir persönlich angen[ehm] aber für das Funktionieren des Programms nicht notwendig. Natürlich sind auch die Alpha-Texte variabel.

2. SIZE-TEST

Obwohl schon in prisma 1 ein Speichergrößentest-Programm abgedruckt ist, hat mein Programm,den Vorteil, daß es (besonders bei großen SIZE-Werten) schneller zum Ziel kommt. Beachtet, daß die Register nicht angezeigt werden, sondern nur die SIZE Größe.

- SF 25
- Solange FS?25=YES
  - X=X+1o
  - VIEW IND X
- Solange FS?25=NO
  - SF 25
  - X=X-1
  - VIEW IND X
- Ausgabe: SIZE=X+1

Das Programm belegt 1o Register. Die früheren Inhalte v[on] T,L, Alpharegister gehen verloren. Das Programm testet erst in 1oer-Schritten, ob die angesprochenen Register existieren, bis es auf ein Nicht-existierendes Register trifft. Dann sucht es rückwärts in 1er Schritten nach de[m] letzten existierenden. Dessen Nummer wird angezeigt.

Übrigens habe ich den SIZE Test der SCI Taste zugeordnet da diese mir das Merken (SIZE-SCI) erleichtert.

Entschuldigt bitte, daß ich nicht mehr Programme mitsch[icke] aber ich habe soeben ein 68o9 uP System gekauft, welche[s] mich voll und ganz beansprucht.

Axel Burkart

```
01◆LBL "SIZE ?"
02 .1911
03 SF 25
04◆LBL 01
05 ISG X
06 VIEW IND X
07 FS? 25
08 GTO 01
09 INT
10◆LBL 02
11 SF 25
12 DSE X
13 VIEW IND X
14 FC?C 25
15 GTO 02
16 ISG X
17 CHS
18 FIX 0
19 "SIZE= "
20 ARCL X
21 AVIEW
22 FIX 2
23 RDN
24 END
```

Achtung: Man setze SIZE gleich 11 oder 21 oder 31 oder 12 oder .....
In jedem Fall kann man hier den gleichen, seltsamen Effekt wie bei de[r] SCROLLING Routine von Norbert Weber und Michel Majerus beobachten !!!!

Oliver

Programm PLOT1: Manuelles Plotten einer gegebenen Funktion mit automatischer Skalierung

Für die graphische Darstellung der Funktion $y=f(x)$ berechnet PLOT1 bei vorgegebener Länge der Abszissen- und Ordinatenachse und ggf. vorgegebener Lage des Ursprungs den Zeichnungsmaßstab, ermittelt die Lage der für die Skalenbeschriftung ausgewählten Achsenpunkte und gibt -gemessen von der linken unteren Ecke des Zeichenfeldes- die Kordinaten der Kurvenpunkte an.

Die Funktion $y=f(x)$ muß zwischen LBL FN und END gespeichert sein; dabei ist FN der Funktionsname (maximal 6 Alpha-Zeichen), das Argument x steht bei Funktionsaufruf im X-Register, wo nach der Funktionsberechnung der Wert y steht.

Eingabeparameter:

Funktionsname FN

Anzahl der Nachkommastellen für die Skalenbeschriftung von Abszisse und Ordinate

Ggf. Koordinaten $u_\emptyset$ bezw. $v_\emptyset$ des Ursprungs im Zeichenfeld

Abszissenlänge a und Ordinatenlänge b des Zeichenfeldes

Extremwerte x_{min}, x_{max}, y_{min} und y_{max} von $y=f(x)$.

Mathematische Grundlagen:

Die Funktion $y=f(x)$ soll auf ein u, v-Koordinatensystem abgebildet werden, dessen Ursprung in der linken unteren Ecke des Zeichenfeldes liegt. Es werden die linearen Abbildungen $x \leftrightarrow u$ und $y \rightarrow v$ gesucht; die folgende Betrachtung wird auf $x \leftrightarrow u$ beschränkt, sie verläuft für $y \rightarrow v$ entsprechend:

Der Bereich von x sei durch x_{min} und x_{max} ($x_{min} < x_{max}$) definiert. Für die u-Achse sind zwei Fälle zu unterscheiden:

1.) Die Lage $u_\emptyset$ des Ursprungs von x ist nicht vorgegeben.
Dann gilt:

$$x = u\frac{x_{max}-x_{min}}{a} + x_{min}$$

$$u = (x - x_{min})\frac{a}{x_{max}-x_{min}} \text{ '.}$$

2.) Die Lage $u_Ø$ des Ursprungs von x ist vorgegeben. Dann muß der Abbildungsmaßstab so angepaßt werden, daß sowohl x_{min} als auch x_{max} noch auf der Abszissenlänge a dargestellt werden können. Je nachdem, ob man x_{min} oder x_{max} zur Berechnung benutzt, gelten die Formelsätze

$$x = -u\frac{x_{min}}{u_Ø} + x_{min}$$

$$u = -(x - x_{min})\frac{u_Ø}{x_{min}}$$

bezw.

$$x = (u - u_Ø)\frac{x_{max}}{a-u_Ø}$$

$$u = x\frac{a-u_Ø}{x_{max}} + u_Ø$$

Wenn $u_Ø = Ø$ oder $-(x_{max} - x_{min})\frac{u_Ø}{x_{min}} > a$ ist, wird der zweite Formelsatz verwendet, andernfalls der erste.

Programmablauf:

1. Programmstart durch XEQ PLOT1
2. Eingabe der Parameter im Dialogverkehr mit R/S. Wenn die Lage des Ursprungs von y=f(x) nicht vorgegeben werden soll, dann wird auf die Frage "UØ ?" bezw. "VØ ?" einfach mit R/S geantwortet.
3. Mit der Anzeige "SKAL. X:A Y:B" ist die Eingabesequenz abgeschlossen. Mit der Eingabe von NN (A) bezw. NN (B) wird die Lage des Punktes NN auf der u- bezw. der v-Achse angegeben, was z. B. zum Beschriften dieser Achsen dienen kann.
4. Soll die Kurve y=f(x) fortlaufend gezeichnet werden, so ist (C) zu drücken. Im Dialog sind dann der Anfangswert u_A und der Zuwachs Δu einzugeben. Wird auf die Frage nach der Startabszisse u_A mit R/S geantwortet, so wird dieser Wert automatisch auf Ø gesetzt.
   u(x) und v(y) werden nun -jeweils nach Drücken von R/S- fortlaufend angezeigt.
5. Sollen die zu bestimmten einzelnen u-Werten gehörenden v-Werte angezeigt werden, so ist nach Eingabe des u-Wertes (D) zu drücken. Dieser Vorgang kann auch beliebig oft während der Prozedur 4 durchgeführt werden.

Speicherplan:

RØØ: Funktionsname FN

RØ1: Nachkommastellen für Abszissenbeschriftung

RØ2: Nachkommastellen für Ordinatenbeschriftung

RØ3: ggf. $u_Ø$ Lage des Ursprungs von y=f(x)

RØ4: ggf. $v_Ø$

RØ5: Abszissenlänge a

RØ6: Ordinatenlänge b

RØ7: x_{min}

RØ8: x_{max}

RØ9: y_{min}

R1Ø: y_{max}

R11: $\frac{x_{max}}{a-u_Ø}$ bezw. $-\frac{x_{min}}{u_Ø}$ bezw. $\frac{x_{max}-x_{min}}{a}$

R12: $\frac{y_{max}}{b-v_Ø}$ bezw. $-\frac{y_{min}}{v_Ø}$ bezw. $\frac{y_{max}-y_{min}}{b}$

R13: Startabszisse u_A

R14: Abszissenzuwachs Δu

R15: Zwischenspeicher

Flags:

Es werden die Flags ØØ ... Ø3 und 22 verwendet.

Programmspeicherbedarf:

PLOT1 benötigt 61 Programmspeicherstellen; daher wird in den meisten Fällen -außer, wenn das Programm für f(x) sehr kurz ist- ein zusätzliches Memory Module benötigt.

Struktogramme:

PLOT1

Hauptprogramm

Flags ØØ ... Ø3 löschen

Parameter einlesen und abspeichern

Skalenfaktoren berechnen

(A) Beschriftung der Abszisse

(B) Beschriftung der Ordinate

(C) fortlaufende Kurvenberechnung

(D) Berechnung von Einzelpunkten der Kurve

Unterprogramm Ø1

Abbildung x => u

Unterprogramm Ø2

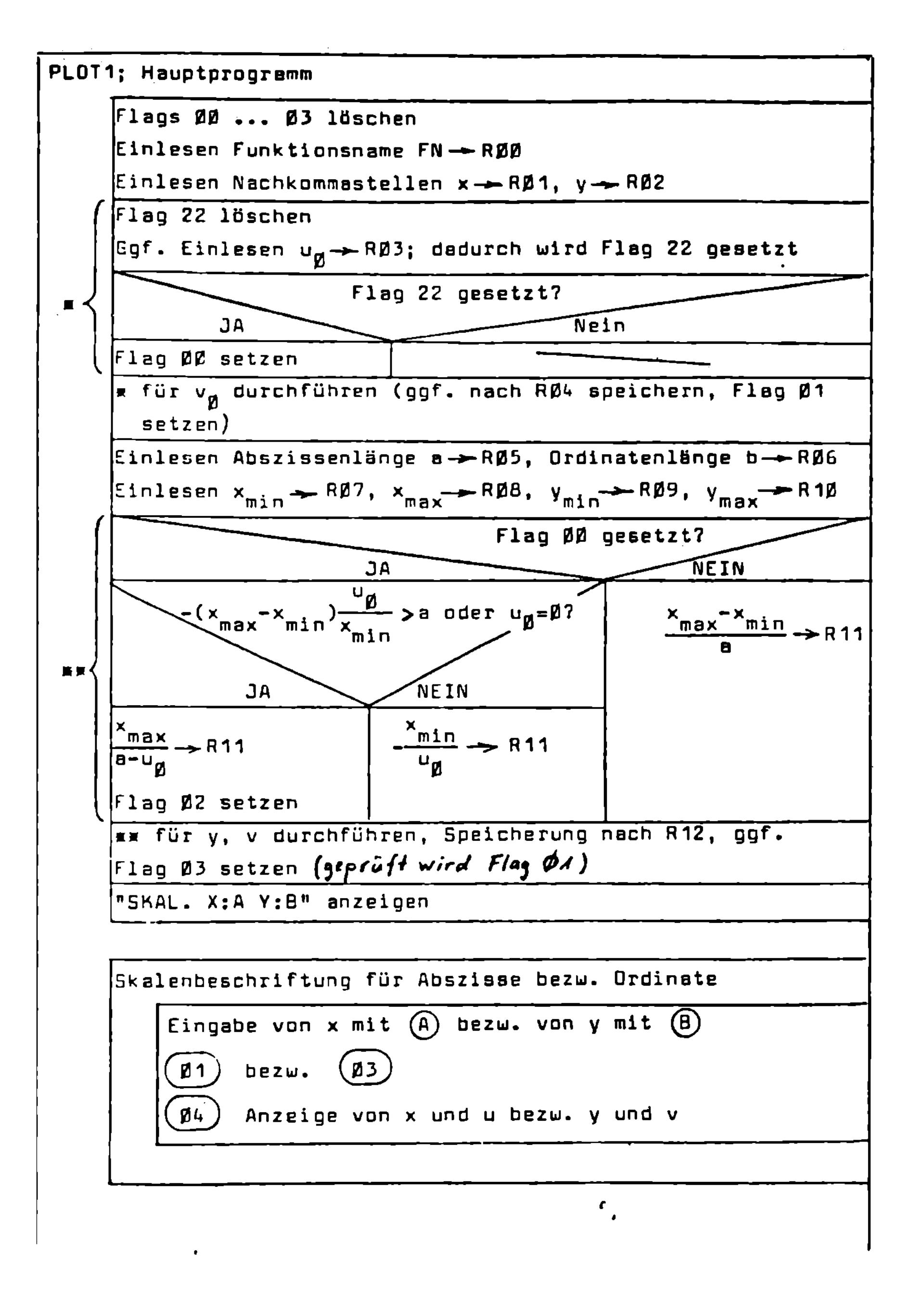
Abbildung u => x

Unterprogramm Ø3

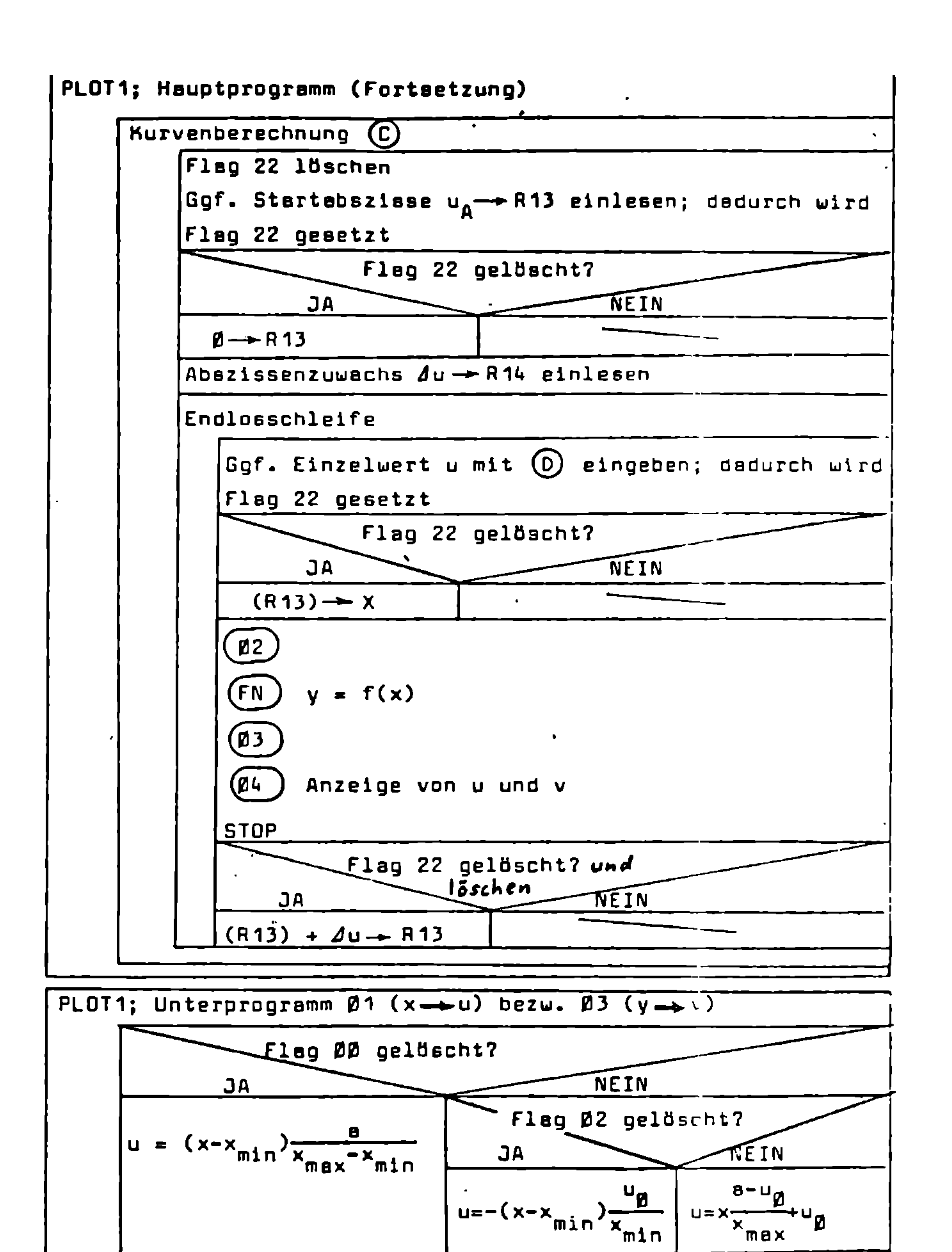
Abbildung y => v

Unterprogramm Ø4

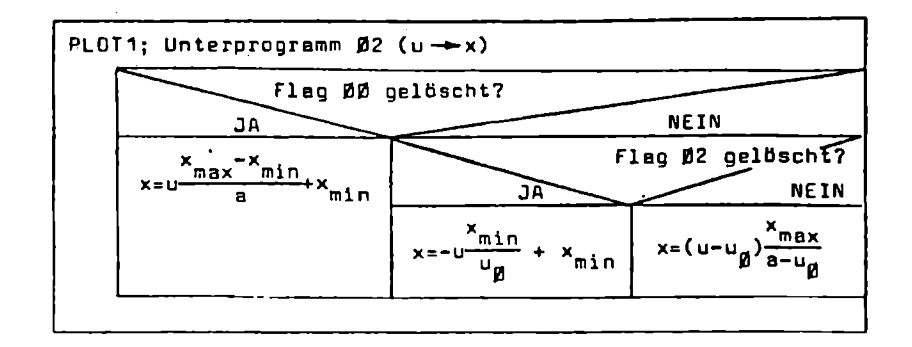
Anzeige von zwei Zahlenwerten (der erste mit wählbarer Nachkommastellenzahl)



PLOT1; Hauptprogramm
Flags ØØ ... Ø3 löschen
Einlesen Funktionsname FN → RØØ
Einlesen Nachkommastellen x → RØ1, y → RØ2
Flag 22 löschen
Ggf. Einlesen u$_Ø$ → RØ3; dadurch wird Flag 22 gesetzt
Flag 22 gesetzt?
JA
Nein
Flag ØØ setzen
* für v$_Ø$ durchführen (ggf. nach RØ4 speichern, Flag Ø1 setzen)
Einlesen Abszissenlänge a → RØ5, Ordinatenlänge b → RØ6
Einlesen x_{min} → RØ7, x_{max} → RØ8, y_{min} → RØ9, y_{max} → R1Ø
Flag ØØ gesetzt?
JA
NEIN
$-(x_{max}-x_{min})\frac{u_Ø}{x_{min}}$ > a oder $u_Ø$=Ø?
JA
NEIN
$\frac{x_{max}-x_{min}}{a}$ → R11
$\frac{x_{max}}{a-u_Ø}$ → R11
Flag Ø2 setzen
$-\frac{x_{min}}{u_Ø}$ → R11
** für y, v durchführen, Speicherung nach R12, ggf. Flag Ø3 setzen (geprüft wird Flag Ø1)
"SKAL. X:A Y:B" anzeigen
Skalenbeschriftung für Abszisse bezw. Ordinate
Eingabe von x mit (A) bezw. von y mit (B)
(Ø1) bezw. (Ø3)
(Ø4) Anzeige von x und u bezw. y und v



PLOT1; Hauptprogramm (Fortsetzung)
Kurvenberechnung (C)
Flag 22 löschen
Ggf. Startabszisse u_A → R13 einlesen; dadurch wird Flag 22 gesetzt
Flag 22 gelöscht?
JA
NEIN
Ø → R13
Abszissenzuwachs Δu → R14 einlesen
Endlosschleife
Ggf. Einzelwert u mit (D) eingeben; dadurch wird Flag 22 gesetzt
Flag 22 gelöscht?
JA
NEIN
(R13) → X
Ø2
FN
y = f(x)
Ø3
Ø4
Anzeige von u und v
STOP
Flag 22 gelöscht? und löschen
JA
NEIN
(R13) + Δu → R13
PLOT1; Unterprogramm Ø1 (x → u) bezw. Ø3 (y → v)
Flag ØØ gelöscht?
JA
NEIN
u = (x-x_min) a/(x_max-x_min)
Flag Ø2 gelöscht?
JA
NEIN
u=-(x-x_min) u_Ø/x_min
u=x (a-u_Ø)/x_max + u_Ø
(Im Unterprogramm Ø3 werden entsprechend die Flags Ø1 und Ø3 geprüft)



Listing:

```
    oLBL"PLOT1"
     CF ØØ
     CF Ø1
     CF Ø2
     CF Ø3
     " FUNKTION ?"
     AON
     PROMPT
     AOFF
Ø1Ø  ASTO ØØ
     "NACHK. X ↑ Y?"
     PROMPT
     STO Ø2
     RDN
     STO Ø1
     CF 22
     "    UØ ?"
     PROMPT
     STO Ø3
Ø2Ø  FS?C 22
     SF ØØ
     "    VØ ?"
     PROMPT
     STO Ø4
     FS?C 22
     SF Ø1
     "LAENGE ABSZISSE"
     PROMPT
     STO Ø5
Ø3Ø  "LAENGE ORDINATE"
     PROMPT
     STO Ø6
     "XMIN ↑ XMAX?"
     PROMPT
     STO Ø8
     RDN
     STO Ø7
     "YMIN ↑ YMAX?
     PROMPT
Ø4Ø  STO 1Ø
     RDN
     STO Ø9
     FS? ØØ
     GTO Ø5
     RCL Ø6
     RCL Ø7
     -
     RCL Ø5
     /
Ø5Ø  STO 11
     GTO Ø6
    oLBL Ø5
     RCL Ø3
     X=Ø?
     GTO Ø7
     RCL Ø7
     RCL Ø8
     -
     *
Ø6Ø  RCL Ø7
     /
     RCL Ø5
     X<Y?
     GTO Ø7
     RCL Ø7
     RCL Ø3
     /
     CHS
     STO 11
Ø7Ø  GTO Ø6
    oLBL Ø7
     RCL Ø8
     RCL Ø5
     RCL Ø3
     -
     /
     STO 11
     SF Ø2
    oLBL Ø6
Ø8Ø  FS? Ø1
```

```
     GTO Ø8
     RCL 1Ø
     RCL Ø9
     -
     RCL Ø6
     /
     STO 12
     GTO Ø9
    oLBL Ø8
Ø9Ø  RCL Ø4
     X=Ø?
     GTO 1Ø
     RCL Ø9
     RCL 1Ø
     -
     *
     RCL Ø9
     /
     RCL Ø6
1ØØ  X<Y?
     GTO 1Ø
     RCL Ø9
     RCL Ø4
     /
     CHS
     STO 12
     GTO Ø9
    oLBL 1Ø
     RCL 1Ø
11Ø  RCL Ø6
     RCL Ø4
     -
     /
     STO 12
     SF Ø3
    oLBL Ø9
     " SKAL. X:A Y:B"
     AVIEW
     SF 27
12Ø  RTN
    oLBL A
     STO 15
     XEQ Ø1
     RCL 15
     RCL Ø1
     XEQ Ø4
     RTN
    oLBL B
     STO 15
13Ø  XEQ Ø3
     RCL 15
     RCL Ø2
     XEQ Ø4
     RTN
    oLBL C
     CF 22
     " STARTABSZ.?"
     PROMPT
     FC?C 22
14Ø  Ø
```

```
     STO 13
     "  ZUWACHS?"
     PROMPT
     STO 14
     CF 22
    oLBL D
     FC? 22
     RCL 13
     STO 15
15Ø  XEQ Ø2
     XEQ IND ØØ
     XEQ Ø3
     RCL 15
     1
     XEQ Ø4
     STOP
     RCL 14
     FC?C 22
     ST+ 13
16Ø  GTO D
     RTN
    oLBL Ø1
     FC? ØØ
     GTO 11
     FC? Ø2
     GTO 11
     RCL 11
     /
     RCL Ø3
17Ø  +
     RTN
    oLBL 11
     RCL Ø7
     -
     RCL 11
     /
     RTN
    oLBL Ø2
     FC? ØØ
18Ø  GTO 12
     FC? Ø2
     GTO 12
     RCL Ø3
     -
     RCL 11
     *
     RTN
    oLBL 12
     RCL 11
19Ø  *
     RCL Ø7
     +
     RTN
    oLBL Ø3
     FC? Ø1
     GTO 13
     FC? Ø3
     GTO 13
     RCL 12
2ØØ  /
```

```
      RCL Ø4                CLA
      +                     FIX IND X
      RTN                   ARCL Y
     oLBL 13                "⊢ "
      RCL Ø9                FIX 1
      -                     ARCL Z
      RCL 12                AVIEW
      /                 218 END
      RTN
 21Ø oLBL Ø4
```

Beispiel:

Die Funktion $y = f(x) = x^2 - x - 1$ soll zwischen $x_{min} = -5$ und $x_{max} = 3$ auf ein Blatt Millimeterpapier, DIN A 4, Hochformat gezeichnet werden. Die Kantenlängen des Millimeternetzes betragen a = 18Ø und b = 28Ø mm. Der Ursprung von $y = f(x)$ soll auf der Abszissenachse bei $u_Ø$ = 1ØØ mm liegen; für die Lage auf der Ordinatenachse wird keine Forderung gestellt.

Die Funktion erhält den Namen PAR:

```
oLBL"PAR"
 X↑2
 LASTX
 -
 1
 -
 END
```

Durch Probieren mittels dieses Programmes, das man zweckmäßigerweise einer Taste zuordnet, erhält man y_{min} = -1.25 und y_{max} = 29. Beim nachfolgenden Dialog mit PLOT1 sind die Benutzereingaben unterstrichen:

<u>XEQ PLOT1</u> FUNKTION ? <u>PAR R/S</u> NACHK. X ↑ Y? <u>1 ↑ Ø R/S</u>
UØ ? <u>1ØØ R/S</u> VØ ? <u>R/S</u> LAENGE ABSZISSE <u>18Ø R/S</u>
LAENGE ORDINATE <u>28Ø R/S</u> XMIN ↑ XMAX? <u>-5 ↑ 3 R/S</u>
YMIN ↑ YMAX? <u>-1.25 ↑ 29 R/S</u> SKAL. X:A Y:B

Die Position der Skalenbeschriftung erhält man mit Ⓐ und Ⓑ. Für den Ursprung erhält man z. B. <u>Ø A</u> Ø.Ø 1ØØ.Ø (wie gefordert) <u>Ø B</u> Ø 11.6 (vom Programm errechnet, da keine Forderung). Mit Ⓒ bezw. Ⓓ erhält man fortlaufend bezw. punktweise die Koordinaten von $y = f(x)$ im Millimeternetz, z. B. für x = Ø: <u>1ØØ D</u> 1ØØ.Ø 2.3; oder für x_{max} = 3: zunächst Ermittlung des zugehörigen Abszissenpunktes: <u>3 A</u> 3.Ø 16Ø.Ø; dann Bestimmung der Koordinaten: <u>16Ø D</u> 16Ø.Ø 57.9 . Die Benutzung von Ⓒ ist auf Seite 2, Punkt 4 erklärt.

Hamburg 73, den 12.Ø6.198Ø K. W. Hoenow
Alaskaweg 18

Dear Richard,

'more bugs in the box? Is XCL actually RCL & STO ?

When lowering the curtain to view program registers as data, you expect to see the first digit of the first byte altered to a "1" because of the normalization process on recalled numbers. (See Bill Wickes· foundamental research V6N8P28a+b). But it is not only that you see that digit changed, the digit actualy has become changed to a "1" in the register itself by the RCL.
The routine UN in the following example doesn't mean anything, it's just 28 bytes (4 registers) to demonstrate the effect. It is to be loaded on top of memory. Routine CR is the curtain raiser (Keith Jarett in V7N4P24).

Roo	1.Byte	(-k, CR,	Roo, k, CR)	⟹					1.Byte
R_{-1}	LBL Co	◇ oo	alpha F3	◇ oo	U 55	N 43	+ 4o		c 1o
R_{-2}	- 41	x 42	/ 43	x<y 44	x>y 45	x≤y 46	S+ 47		1 11
R_{-3}	S+ 48	alpha F6	Q 51	R 52	S 53	T 54	U 55		8 18
R_{-4}	V 56	AVIEW 7E	BEEP 86	R/S 84	E C8	N o3	D o9		ä 16

If you won't believe the RCL is the evil-doer, simply skip the RCL and try the example again; the curtain raiser works ok. So now I must conclude: Because the first digit (the sign digit) is other then o or 1 the recalled number gets normalized (in the example here becomes a 1) and stored back into the register it was called from. So, RCL behaves like RCL & normalize & STO.

The hex-codes in the above example can easily be verified by adding DECODE (s. Wickes in V7N2P35-36) after the curtain has been restored. The following simple routine RR does it for you. Of course, it needs CR and DECODE and a SIZE greater than 5 for DECODE and UN at TOM (top of memory).

The END-statement (C8 o3 o9) links to the next GLOBAL-statement up in memory (see Charles Close in V7N3P9):
C - Global-Statement,
C8 o3 - distance to next GLOBAL up in memory, calculated as Hex(8)=Bin(1ooo) and the first 3 bits give Bin(1oo)=Dec(4) give 4 bytes to the distance; Hex(o3)=Dec(3) add 3 registers to that distance, totalling up to 25 bytes.
o9 - the program has been PACKED.
The addressed GLOBAL-statement 25 bytes up from the location of the END-statement is - in the example here-the LBL UN statement, Co oo linking to TOM.

F3 the alpha-indicator gives room for one byte more than needed for the alphas U and N. The byte after F3 is the place where a possible key-assignment would be stored in modified reverse matrix location (see Wickes in V6N8P28). Hex(oo) indicates no

```
LBL'UN
END        28 BYTES
LBL'CR
END        97 BYTES
.END.      08 BYTES

 01◆LBL "UN"
 02 +
 03 -
 04 *
 05 /
 06 X<Y?
 07 X>Y?
 08 X<=Y?
 09 Σ+
 10 Σ-
 11 "QRSTUV"
 12 AVIEW
 13 BEEP
 14 STOP
 15 END

   -4.
    XEQ "CR"
      RCL 03
  "s"    ***
      RCL 02
"BCDEFG"
         ***
      RCL 01
"ᵥQRSTU"
         ***
      RCL 00
"ΣΓαH←σ"
         ***
    4.
    XEQ "CR"

          CAT 1
0
END        28 BYTES
LBL'CR
END        97 BYTES
.END.      08 BYTES

          CAT 1
0

→ 01 0
→ 02 "◆UN"
  03 +
→ 04 1
  05 *
  06 /
  07 X<Y?
  08 X>Y?
  09 X<=Y?
  10 Σ+
→ 11 8
→ 12 "QRSTUä"
  13 AVIEW
  14 BEEP
  15 STOP
  16 END
```

→

assignment to a user key ha[illegible] made for UN.

```
           CAT 1
LBL'RR
END        32 BYTES

 01◆LBL "RR"
 02 STO Y
 03 XEQ "CR"
 04 RCL 00
 05 VIEW X
 06 X<>Y
 07 CHS
 08 XEQ "CR"
 09 XEQ "DEC
ODE"
 10 END

         -1.
          XEQ "RR"
               "S"
10/00/F3/00/55/4E/40
         -2.
          XEQ "RR"
          "BCDEFG"
11/42/43/44/45/46/47
         -3.
          XEQ "RR"
          "vGRSTU"
18/F6/51/52/53/54/55
         -4.
          XEQ "RR"
          "ΣΓαH+α"
16/7E/86/84/C8/83/09
```

Thanks to Wickes PUC-2 simulator (see V7N3P7) we are able to store any code in (almost) any register. What we can't do, until now, is to see the code of any desired register. The byte-jumper (Wickes in V7N4P26) helps a bit, but you still can't see a program-register without shifting the code by insertion of the "jump-number-code" and so destroying the program itself, equivalently to the method described here by using RCL to see a program register. Don-t the grand-masters of the Synthetic Codes see a way out?

Happy programming! Gerhard Goder (2105)

Münster (West-Germany) Aug, 24th 1980

Gerh Goder

GERH. GODER 2105
INST. F. MED. INFORMATIK
HUEFFERSTR. 75
4400 MUENSTER, GERMANY

12-80

Dieser Brief wird nur PPC Mitgliedern verständlich sein.
Dennoch: Wer kann Gerhard helfen ??

Oliver

'/

WAS IST DAS ? P P C ?

Was der PPC den sei - eine Frage, die mir oft gestellt wurde. Nun- es ist der größte Taschenrechnerclub der Welt für HP Rechner und hat gegenwärtig 2200 Mitglieder (nicht 5500 wie irrtümlich von mir letzten Monat behauptet). Es sind nicht nur echte HP-Profis dabei, die sich in Eigenarbeit Interfaces gebaut haben (!) und diese bald (?) kommerziell verkaufen wollen, sondern auch "wilde" Programmierer, die die gesammte synthetische Programmierung erst entdeckt haben. "Echte" Programme sind weniger zu finden. Die Clubzeitschrift erscheint monatlich. Der Beitrag wurde vor kurzem erhöht und dürfte jetzt bei 35 bis 40 US $ liegen. Wer einmal ein PPC-Journal in die Finger bekam, weiß, daß es praktisch unmöglich ist, es in unserem Club auf dem Gebiet der synthetischen Programmierung soweit zu bringen, wie der PPC, geschweige denn, diesen zu überholen. Es sind halt Profis. Hunderte.
Deshalb möchte ich das Kapitel der synthetischen Programmierung in unserem Club nicht weiter ausdehnen. Wer sich für diese letztlich reinen Spielereien interessiert, der möge sich an Richard Nelson, Editor PPC Journal, 2541 West Camden Place, Santa Ana, California 92704, USA wenden und (auf Englisch !) um ein "Special Issue" bitten. Der Anforderung sollten zwei internationale Antwortscheine (bei jedem Postamt) beiliegen. Ein kostenloses Probeexemplar kommt dann per Luftpost mit weiteren Einzelheiten.
Ich sage dies mit reinem Gewissen, denn unseren Club werde ich damit sicher nicht schädigen. Wir wollen nicht den Übersetzer des PPC spielen, dafür gibt es schon andere. Wir wollen viel mehr eigene Leistungen vollbringen und zu den synthetischen "Sachen" im PPCJ höchstens einige Ergänzungen liefern. Noch eine Warnung zum Schluß: Wer in den PPC eintreten will, der sollte viel, viel Zeit mitbringen; es gibt viel zu verstehen. Einige hundert Seiten bisheriger Journale müssen nachbestellt werden.

Im Zeit[illegible] des Energiesparens ist jedermann an der Kenntnis der Leistu[illegible]aufnahme elektrischer Geräte interessiert. Bei vorhandensein einer Stoppuhr und eines kurzen Programms bietet sich der Zähler als Leistungsmesser an.

Vorgang der Messung: Es werden n Umdrehungen der Läuferscheibe des Zählers gestoppt. Dazu kann die, im Programm enthaltene Stoppuhr dienen.

Für Nichtelektriker: bei normalen Haushaltzählern ist C=1 zu setzen. Kp ist am Leistungsschild des Zählers in U/kWh vermerkt.

Kq und cos dient nur zur Ermittlung der Blindleistung bei Industrieanlagen.

C = Wandlerkonstante
Kp= Zählerkonstante Wirkarbeit kWh/U
Kq= Zählerkonstante Blindarbeit in kVArh/U
T = Zeit für N Umdrehungen

```
01•LBL "ZAESTO"
02 5
03 "C="
04 XEQ 00
05 1
06 "KP="
07 XEQ 00
08 2
09 "KQ="
10 XEQ 00
11•LBL A
12 CF 22
13 "T=?"
14 PROMPT
15 FC? 22
16 XEQ B
17 "N=?"
18 PROMPT
19 FIX 2
20 "      P Q"
21 PROMPT
22•LBL C
23 "P="
24 AVIEW
25 RCL 01
26 XEQ 03
27 STO 03
28 RCL 04
29 X<>Y
30 RTN
31 GTO 01
32•LBL D
33 "Q="
34 AVIEW
35 RCL 02
36 XEQ 03
37 STO 04
38 RCL 03
39 X<>Y
40 RTN
41•LBL 01
42 "           COS"
43 PROMPT
44 GTO A
45•LBL E
46 RCL 04
47 RCL 03
48 R-P
49 X<>Y
50 COS
51 "COS="
52 ARCL X
53 AVIEW
54 RTN
55 GTO A
56•LBL 03
57 RCL Y
58 X<>Y
59 /
60 R↑
61 /
62 3600
63 *
64 RCL 05
65 *
66 ARCL X
67 AVIEW
68 RTN
69•LBL 00
70 STO 90
71 RCL IND 00
72 ARCL X
73 PROMPT
74 STO IND 00
75 RTN
76•LBL B
77 FIX 1
78 ,21102
79 FC?C 00
80 STO 00
81 RCL 00
82 ENTER↑
83 ENTER↑
84 ENTER↑
85 CLX
86 "R/S, ON"
87 PROMPT
88 SF 11
89•LBL 02
90 VIEW X
91 +
92 FS? 11
93 GTO 02
94 PSE
95 RTN
96•LBL J
97 SF 00
98 "T-SOLL?"
99 PROMPT
100 /
101 ENTER↑
102 ENTER↑
103 1
104 -
105 100
106 *
107 CLA
108 ARCL X
109 "-%"
110 PROMPT
111 RDN
112 ST/ 00
113 GTO B
114 END
LBL'ZAESTO
END          237 BYTES
```

Programmstart: XEQ "ZAESTO"
Konstanten nach Aufruf eingeben. Bei 'T=?' Zeit eingeben, wenn diese mit getrennter Uhr ermittelt wurde. Ansonst nur R/S drücken. Es erfolgt Sprung in Unterprogramm Uhr. (Siehe unten)
Nach Eingabe von N zeigt der Rechner die weitere Tastenzuordnung an.

Berechnung von P: C
Berechnung von Q: D
Berechnung des Leistungfaktors: E

Nach Berechnung von Q (P) steht der zuvor berechnete Wert im Y Register. So können mehrere Leistungswerte mit Σ + summiert werden.

Stoppuhr: LBL: B
Start mit R/S
Stopp mit ON
Nach Wiedereinschalten des Rechners wird die Zeit kurz angezeigt und es erfolgt Rücksprung ins Hauptprogramm.

Justieren der Stoppuhr:
Zeit stoppen (ca 200sek), J, richtige Zeit eingeben, R/S, Fehler in%, R/S, Vorgang wiederholen. Wenn Zeit stimmt den Wert im R00 bei Programmschritt 78 festhalten. Uhrenprogramm immer an 1.Stelle im Speicher eingeben und packen!

HALLO HP-41C BESITZER

Vielleicht störte Sie auch schon die Tatsache , dass Ihr 4I c mit Verbindung des Printers eigentlich nur drei Speichermodule fassen kann.Um dies zu "beheben",habe ich folgenden Umbau unternommen.Ich hatte nach diesem Eingriff noch keine Schwierigkeiten oder eventuelle Pannen.

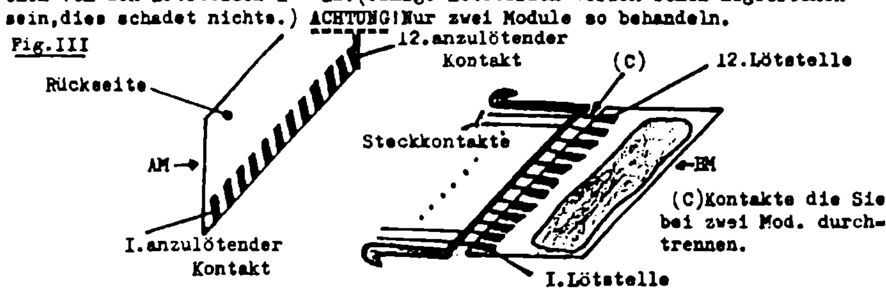
A↑ B↓ **Fig.I**

Sie können Ihre Modulhülle "aufbrechen",indem Sie die Plastikkanten rechts oder links in die Pfeilrichtung (A B)auseinanderreissen.(Wenn möglich nur mit den Fingern versuchen,nicht mit einem Gegenstand aufbrechen).Eine Seite wird sich höchstwahrscheinlich lösen,danach können Sie die andere Seite abheben.Sie werden in gleicher Weise mit den anderen Modulen verfahren.Es kann vorkommen,dass Sie ein oder zwei Module nicht geöffnet bekommen,ohne die Hülle zu zerstören.Mit einer kleinen Säge können Sie dann diese Mod. längsseits aufschneiden.Zwei ganze hüllen müssen Sie auf jeden Fall behalten.

Fig.II Vorderseite Rückseite

Danach schneiden oder sägen Sie die Kontakte von I bis I2 mit einem feinen Instrument durch.Sie entfernen dann anhand eines Lötkolbens die Metallstückchen von den Lötstellen I - I2.(einige Lötstellen werden schon abgebrochen sein,dies schadet nichts.) ACHTUNG!Nur zwei Module so behandeln.

Fig.III



Dann werden Sie die losgeschnittenen Mod. an die noch ganzen Module anlöten. Beachten Sie folgendes Schema und die folgende Tabelle:

BM	AM
I	I
3	3
5	5
7	7
9	9
II	II
3	2
4	4
6	6
8	8
Io	Io
I2	I2

BM = Baisimodul auf das das AM = aufzulötende Modul aufgesetzt wird.Verlöten Sie die numerierten Lötstellen von BM mit denen von AM.

Waschen Sie dann das eventuell vorhandene Lötfett mit Seife und einer feinen Bürste ab; trocknen Sie das MODUL und biegen Sie die neue Platte auf die alte. Kontrollieren Sie ob keine Drähte sich zufällig berühren.Setzen Sie die Plättchen in das Modulgehäuse,ohne dieses wieder zusammenzukleben.

Dann set[illegible]ie die zwei neuen Modulgehäuse in PORT I und PORT II. Probieren Si[illegible]lbst! XEQ SIZE 000 , dann PRGM .Erscheint in der Anzeige nicht 00 REG 319 ,so wechseln Sie die beiden Kammern,und Sie wiederholen die gleiche Tastenfolge.Im Prinzip müsste es klappen.Sie haben nun noch PORT 3 und PORT 4 frei,um eventuell den Printer und ein vorprogrammiertes Modul einzufügen.Viel Spass!!!

P.S: Falls Schwierigkeiten auftreten,so können Sie mich schriftlich oder telefonisch (nach 19^{00}) erreichen,um das Problem zu lösen.

Weber Norbert Tel: 444747
147,Val Ste. Croix
LUXEMBOURG

Tschüss
Web (735)

ACHTUNG !

Wer sich nicht traut, obigen Umbau selbst vorzunehmen, kann auch zwei Module (per Wertbrief!) an Norbert schicken. Er hat sich bereit erklärt, diese für Mitglieder gegen eine Gebühr von 50,-- DM umzubauen. In diesen 50,-- DM ist nicht nur Rückporto, sondern auch eine "Versicherung" gegen fehlgeschlagenen "Umbauversuch" enthalten. Der Besitzer der umzubauenden Module nimmt also kein Risiko auf sich.
Damit wird auch "Anti-Elektronikern" ermöglicht, vier Memory Module, d.h. zwei Double Density Module (DDM), wie die umgebauten Module genannt werden, zusammen mit Kartenleser und Drucker oder ROM's gleichzeitig zu betreiben.

Lieber Oliver !

Probier mal folgendes Programm. Du kannst den Alpha Text in Zeile 2 beliebig ändern. Übrigens: Drücke mal während des Programmablaufs irgendeine Taste, außer ON und R/S.

Beste Grüße Tschüss

Michel Majerus und
Norbert Weber

```
LBL A
"SCROLLING"
o
AVIEW
LBL o4
SF 25
1/X
GTO o4
```

!!!

Programmvergütung ? NEIN -- Beitragsvergütung ? JA (einfach supe

Schluß - Aus und Vorbei mit der Programmvergütung. Keine kostenlose Programme meh für eingesandte. Ich habe die Programmbibliothek aufgelöst, nahezu alle Mitgliede stimmten mir zu. Aber was statt dessen ? Um den Anreiz, Beiträge zu unserer Clubzeitschrift(chen) zu schreiben* werden wir jeden Oktober (ab 1981) eine Wahl der b ten Artikel durchführen. Nun will ich hier keine großartigen Versprechungen leist was die Preise anbetrifft, aber soviel sei gesagt:

* zu steige

1. Preis: 1 Modul für den HP-41C nach freier Wahl

Was sonst noch alles dazukommt, hängt ganz von der derzeitigen Finanzlage ab.
Es wird sich aber sehen lassen können !! Oliver

HP85

Im Januar 19[illegible] wagte die Firma HEWLETT-PACKA[illegible] [illegible]en Schritt zu den Personal Computer. Der '85 ist wieder eines dieser vielgerühmten und bewunderten Meisterwerke der amerikanischen Weltmarke geworden.

Ich hatte im Sommer dieses Jahres die Gelegenheit, den 85er zwei Tage lang zu testen. Hier mein Bericht über

DER BRUDER DES 41ERS

Als ich den stabilen Transportkoffer öffnete, errinnerte mich dieses Gerät, das da zum Vorschein kam, zuerst an meine HERMES-Schreibmaschine. Es sah nähmlich ebenso technisch perfekt aus. Doch da waren einige kleine, doch sehr frappante Unterschiede: Meine Schreibmaschine hat zum Beispiel keinen Bildschirm.





Ohne lange die recht umfangreiche Bedienungsanleitung zu befragen, griff ich zum Netzkabel und schloss das Gerät an. Und siehe da: schon nach einigen Momenten war das System bereit. Es konnte also sofort los gehen!

Ich steckte die beiliegende Demo-Cassette in den Tape Drive und sogleich surrte der fast lautlose Thermodrucker los und zeichnete ein Selbstbildnis.Damit aber noch nicht genug: Hypothekenzins per Graphik, dasselbe in tabelarischer Form und der ganze Zeichensatz folgen. Als Höhepunkt spielt der eingebaute Tongenerator noch Mozart!

Soweit, sogut. Doch dieses etwa SFr. 7'200.-- (ca.) teure Gerät ist natürlich kein Spielcomputer. Im HP-85A (so die richtige Bezeichnung), sind alle guten Eigenschaften der heutigen Personal Computer soweit möglich vereinigt. Hier nun einige technische Daten:

Benutzerspeicher (Standard) .... 16 K Byte
Rechenbereich .... +/- 9. E-11 * +/- 10↑499
Gewicht .... 9.06 kg
Ausführungszeit (math. Funktionen/ Ø).. min. (ABS) ... 0.83 MS
(MS = Millisekunden) max. (SEC) .. 51.72 MS

Es besteht ausserdem die Möglichkeit der Speichererweiterung um weitere 16 K Byte (HP-82903A).

- **Drucker**

Druckmethode	:	beweglicher Thermodruckkopf
Papierbreite	:	108 mm
Druckgeschwindigkeit	:	2 Zeilen pro Sekunde
Graphische Auflösung	:	2.63 Punkte/mm

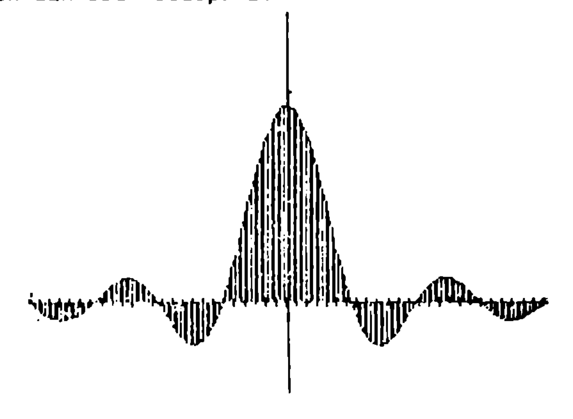
- Bildschirm

Bildschirmgrösse	:	127 mm Diagonale
Kapazität	:	16 x 32 Zeichen oder 192 x 256 Punkte
Matrix	:	5 x 7 Punkte

- Tape Drive

Kapazität	:	210 K (Daten) oder 195 K (Programme)
L/S-Geschwindigkeit	:	254 mm/sec
Zugriffszeit (Ø)	:	9.3 Sekunden
Bandlänge	:	43 Meter

Das Programmieren auf dem 85er macht richtig Spass. Die Sprache des HP-85A ist ein erweitertes BASIC, das alle Anweisungen des Standard ANSI-BASIC enthält. Die Schreibmaschinentastatur wird durch ein numerisches Tastenfeld sowie Steuer-, Editier- und acht Sonderfunktionstasten ergänzt. Die Belegung der Sonderfunktionstasten ist vom Benutzer frei definierbar. Mit Hilfe dieser Tasten sind Programmverzweigungen während des Programmablaufs möglich.
Beim eigendlichen Programmieren wird jede Programmzeile, die mit dem Begriff "Line End" abgeschlossen werden, auf den Syntax hin geprüft, so dass Tippfehler fast ausgeschlossen sind.
Hier nun noch ein Druckbeispiel:



Hier beweist der 85er seine unwahrscheinliche Genauigkeit und Präzition
Fragen bitte an HP richten.

Guido S t u d e r

Niels Nöhren
Kielort 16
2000 Norderstedt
F: (040) 524 28 88

Norderstedt 18.10.1980

Herrn
Oliver Rietschel
Hewlett-Packard Anwender-Club
Postfach 373

2420 Eutin

Liebe Clubfreunde!

Das Programm Bruchrechnung von Clemens Mirgel (vergl. prisma 26/9-80) hat mich richtig begeistert. Ich habe es gleich in meine Programmsammlung aufgenommen. Allerdings stellte ich fest, daß sich das Programm an einigen Stellen verbessern läßt. Ich stelle Euch hier eine optimierte Version vor. Die wesentlichen Veränderungen befinden sich in den Programmteilen =E3=, "KGV" und "K".

Weiterhin möchte ich Euch mein Programm "Lottozahlen" vorstellen. Es liefert in einem Durchgang je 6 verschiedene Zahlen zwischen 1 und 49. Zuerst wird eine Zahl nach der anderen erzeugt, auf Null und Gleichheit überprüft und abgespeichert. Anschließend wird eine nach der anderen angezeigt.

Das Programm wird mit XEQ "L" gestartet. Mit der Anzeige "QUELLZAHL ?" fragt der Rechner nach einer Ausgangszahl, die zwischen Ø und 1 liegen sollte. Mit TONE Ø wird die erste Zahl angekündigt. Mit TONE 9 macht der Rechner auf die 6. Zahl aufmerksam. Für jede weitere Zahlenreihe muß die Taste A gedrückt werden.

Außerdem stelle ich mein Kalenderprogramm vor. Es beruht zwar auf dem Kalenderprogramm der Standardsammlung, ist aber von mir in einigen Punkten verändert worden. Es arbeitet korrekt im gesamten Bereich des Gregorianischen Kalenders und überprüft jedes eingegebene Datum auf seine Legalität.

...

Im Grunde erfüllt das Programm den gleichen Zweck wie das Standardprogramm. Es werden aber nur 4 Register zur Zwischenspeicherung benötigt, alles andere spielt sich ausschließlich im Stack ab.

Alle notwendigen Angaben müssen gleich eingegeben werden, bevor das Programm gestartet wird. Also beispielsweise Datum A, ENTER, Datum B, XEQ "R" zur Berechnung der Tagesanzahl zwischen 2 Daten. Datum A, ENTER, Anzahl der Tage, XEQ "N" zur Ermittlung eines neuen Datums und schließlich: Datum, XEQ "Q" für den entsprechenden Wochentag.

Wird das Datum 29.02.1900 eingegeben, ein Datum, das es nicht gab, weil das Jahr 1900 kein Schaltjahr war, reagiert der Rechner mit der Anzeige "DATUMSFEHLER". Zur Erinnerung steht im X-Register noch einmal das unmögliche Datum.

Ich habe das Programm so ausgelegt, daß die bei uns übliche Datumsform DD,MMYYYY verwendet werden muß.

Für diesmal möchte ich schließen. Es grüßt Euch und wünscht Happy Programming

Euer

Nick Nöl

Anlage: 3 Programmlistings

Programm Bruchrechnung (optimiert):

```
1 LBL BR
2 FIX Ø
3 CF 29
4 Ø
5 GANZZAHL 1?
6 PROMPT
7 STO ØØ
8 "ZAEHLER 1?"
9 PROMPT
10 STO Ø1
11 "NENNER 1?"
12 PROMPT
13 STO Ø2
14 LBL Ø1
15 RCL ØØ
16 RCL Ø2
17 x
18 ST+ Ø1
19 Ø
20 "GANZZAHL 2?"
21 PROMPT
22 STO Ø3
23 "ZAEHLER 2?"
24 PROMPT
25 STO Ø4
26 "NENNER 2?"
27 PROMPT
28 STO Ø5
29 RCL Ø3
30 x
31 ST+ Ø4
32 "- + x / ?"
33 PROMPT
34 LBL Ø2
35 XEQ EB
36 XEQ K
37 RCL Ø1
38 X=Ø?
39 GTO Ø3
40 XEQ ANZ
41 LBL Ø3
42 " "
43 ARCL ØØ
44 AVIEW
45 STOP
46 LBL Ø4
47 RCL Ø8
48 STO Ø2
49 GTO Ø1
50 LBL +
51 XEQ AS
52 ST+ Ø1
53 GTO Ø2
54 LBL -
55 XEQ AS
56 ST- Ø1
57 GTO Ø2
58 LBL x
59 RCL Ø4
60 STx Ø1
61 RCL Ø2
62 RCL Ø5
63 x
64 STO Ø8
65 GTO Ø2
66 LBL /
67 RCL Ø5
68 STx Ø1
69 RCL Ø2
70 RCL Ø4
71 x
72 STO Ø8
73 GTO Ø2
74 LBL EB
75 Ø
76 STO ØØ
77 RCL Ø1
78 RCL Ø8
79 X>Y?
80 RTN
81 /
82 INT
83 STO ØØ
84 RCL Ø1
85 RCL Ø8
86 MOD
87 STO Ø1
88 RTN
89 LBL ANZ
90 " "
91 ARCL ØØ
92 "⊢ "
93 ARCL Ø1
94 "⊢/"
95 ARCL Ø8
96 AVIEW
97 STOP
98 GTO Ø4
99 LBL GGT
100 LBL Ø5
101 MOD
102 LASTX
103 X<>Y
104 X≠Ø?
105 GTO Ø5
106 RDN
107 RTN
108 LBL KGV
109 STO Z
110 X<>Y
111 STx Z
112 XEQ GGT
113 /
114 RTN
115 LBL K
116 RCL Ø1
```

Programm Bruchrechnung

```
117 RCL Ø8
118 XEQ GGT
119 ST/ Ø1
120 ST/ Ø8
121 RTN
122 LBL AS
123 RCL Ø2
124 RCL Ø5
125 XEQ KGV
126 STO Ø8
127 RCL Ø2
128 /
129 STx Ø1
130 RCL Ø8
131 RCL Ø5
132 /
133 STx Ø4
134 RCL Ø4
135 END
```

Programm L[illegible]zahlengenerator

```
 1  LBL L
 2  CF 27
 3  FIX Ø
 4  CF 29
 5  1.ØØ6
 6  STO Ø7
 7  CLX
 8  "QUELLZAHL ?"
 9  PROMPT
10  SF 27
11  X=Ø?
12  GTO A
13  STO ØØ
14  LBL A
15  RCL Ø7
16  ENTER
17  CLX
18  LBL Ø2
19  STO IND Y
20  ISG Y
21  GTO Ø2
22  LBL Ø1
23  RCL ØØ
24  9821
25  x
26  Ø,211327
27  +
28  FRC
29  STO ØØ
30  5Ø
31  x
32  INT
33  X=Ø?
34  GTO Ø1
35  LBL Ø3
36  RCL Ø7
37  X<> Y
38  LBL Ø4
39  RCL IND Y
40  X=Ø?
41  GTO Ø5
42  X=Y?
43  GTO Ø1
44  RDN
45  ISG Y
46  GTO Ø4
47  LBL Ø5
48  RDN
49  STO IND Y
50  ISG Y
51  GTO Ø1
52  RCL Ø7
53  TONE Ø
54  LBL Ø6
55  6,ØØ6
56  X<> Y
57  X=Y?
58  TONE 9
59  RCL IND X
60  STOP
61  RDN
62  ISG X
63  GTO Ø6
64  LBL CLR
65  CF 27
66  SF 29
67  FIX 2
68  CLA
69  CLST
70  -
71  OFF
72  GTO CLR
73  END
```

Programmlisting Kalenderberechnung

```
LBL U, STO Ø4, XEQ Y, X<>Y, X=Ø?, GTO W, CLX, 1, X<>Y, X=Ø?,
GTO W, 12, X<Y?, GTO W, CLX, 1, +, RCL Z, 1 E4, /, +, 1 E2,
/, +, XEQ V, STO Ø5, RCL Ø4, XEQ V, ENTER, X<> Ø5, X<=Y?,
GTO W, X<>Y, +, CLST, RCL Ø4, LASTX, RTN,

LBL W, CLST, RCL Ø4, FIX 6, "DATUMSFEHLER", PROMPT,

LBL Y, INT, ST- L, LASTX, 1 E2, x, INT, ST- L, LASTX, 1 E4,
x, X<>Y, RDN, X<>Y, R↑, RTN,

LBL V, XEQ Y, 1, +, 1/X, Ø,7, ST+ Y, X<> L, X<>Y, CHS, XEQ O,
3Ø,6, x, INT, +, X<>Y, 1 E2, /, INT, 7, ST- Y, CLX, Ø,75,
STx Y, X<> L, X<>Y, INT, ST- Z, CLX, 36525, x, INT, +, RTN,

LBL O, INT, ST+ T, X<>Y, INT, CLX, 12, STx Y, X<> L, X<>Y,
-, RTN,

LBL N, STO Ø1, RDN, XEQ U, RCL Ø1, +, GTO P,

LBL R, STO ØØ, X<>Y, XEQ U, X<> ØØ, XEQ U, RCL ØØ, -, INT,
CLST, FIX 2, LASTX, RTN,

LBL P, ENTER, ENTER, ENTER, 365,25, ST/ Y, X<>Y, 1 E2, /,
INT, 7, -, Ø,75, x, INT, X<>Y, RDN, +, R↑, ENTER, ENTER,
/, 122,1, x, ST- Z, CLX, +, ST/ Y, X<>Y, INT, STx Y, RDN?
INT, -, R↑, X<>Y, ENTER, ENTER, 3Ø,6, ST/ Y, X<>Y, INT,
STx Y, RDN, INT, -, R↑, INT, 14, ST/ Y, X<> L, X<>Y, XEQ O,
X<>Y, 1, X<>Y, %, -, -, ENTER, INT, ST- Y, 1 E2, STx Z,
FIX Ø, CF 29, CLA, ARCL Z, "⊢.", ARCL Y, "⊢-", ARCL T, /,
+, X<>Y, 1 E6, /, +, FIX 6, SF 29, AVIEW, RTN,

LBL Q, XEQ U, 7, MOD, XEQ IND X, CLST, RCL Ø4, FIX 6, AVIEW,
RTN,

LBL ØØ, "SONNTAG", RTN,

LBL Ø1, "MONTAG", RTN,

LBL Ø2, "DIENSTAG", RTN,

LBL Ø3, "MITTWOCH", RTN,

LBL Ø4, "DONNERSTAG", RTN,

LBL Ø5, "FREITAG", RTN,

LBL Ø6, "SAMSTAG", END.
```

Beispiele zur Verwendung des Kalenderprogramms.

1.) Wochentag

Eingabe: 29,021900 Programmstart: XEQ "Q" Resultat: Anzeige: "DATUMSFEHLER", X-Register: 29,021900.

2.) Wochentag

Eingabe: 28,021900, Programmstart: XEQ "Q", Resultat: Anzeige: "MITTWOCH", X-Register: 28,021900/

3.) Anzahl von Tagen zwischen zwei Daten.

Eingabe: 30,021980, Taste: ENTER, Display: 30,02 (bei FIX 2)
Eingabe: 30,091980, Programmstart: XEQ "R", Display: "DATUMSFEHLER", X-Register: 30,021980.

4.) Anzahl von Tagen zwischen zwei Daten.

Eingabe: 29,021980, Taste: ENTER, Display: 29,02,
Eingabe: 30,091980, Programmstart: XEQ "R", Display: 214,00 (Tage).

5.) Berechnung eines zukünftigen Datums.

Eingabe: 19,101980 (Datum), Taste: ENTER, Display: 19,10,
Eingabe: 500 (Tage), Programmstart: XEQ "N",
Display: "3-3-1982", X-Register: 3,031982.

6.) Berechnung eines vergangenen Datums.

Eingabe: 19,101980(Datum), Taste: ENTER, Display: 19,10,
Eingabe: x 500, Taste: CHS, Display: -500,00,
Programmstart: XEQ "N", Display: "7-6-1979", X-Register: 7,061979.

```
MAGISCHE
ZAHLEN
-------------
COPYRIGHT BY R.KLAUC AND
HP-ANW.CLUB, O.RIETSCHEL
-------------
SIZE: 053
START: XEQ MAGIC
EINGABE.UHRZEIT(,SSMMHH)
ODER ZUFALLSZAHL 0<X<1.
-------------
SYSTEM-KONFIGURATION:
 1 MEMORY MODULE
 PRINTER
-------------
```

```
 01♦LBL "MAG
IC"
 02 FIX 0
 03 CF 29
 04 SF 12
 05 ADV
 06 "UHRZEIT
,SMH?"
 07 PROMPT
 08 STO 00
 09 1.05
 10 STO 51
 11 "MAGISCH
E"
 12 PRA
 13 "       ZA
HLEN"
 14 PRA
 15 ADV
 16 CF 12
 17♦LBL 02
 18 1.01
 19 STO 52
 20 " "
 21♦LBL 00
 22 RCL 00
 23 9821
 24 *
 25 .211327
 26 +
 27 FRC
 28 STO 00
 29 10
 30 *
 31 INT
 32 X=0?
 33 GTO 00
 34 STO IND
51
 35 ARCL X
 36 "├ "
 37 ISG 52
 38 SF 00
 39 FC?C 00
 40 XEQ 01
 41 ISG 51
 42 GTO 00
 43 ADV
 44 "SUCHEN
SIE IHRE"
 45 PRA
 46 "SCHLUES
SELZAHL "
 47 "├X AN D
ER"
 48 PRA
 49 "Y. STEL
LE."
 50 PRA
 51 "WENN SI
E "
 52 "├FERTIG
SIND"
 53 PRA
 54 "DRUECKE
N SIE "
 55 "├RUN."
 56 PRA
 57 ADV
 58 ASHF
 59 ASHF
 60 PROMPT
 61 1
 62 STO 51
 63♦LBL 05
 64 RCL IND
51
 65 STO 52
 66 ST+ 51
 67 50
 68 RCL 51
 69 X<=Y?
 70 GTO 05
 71 RCL Z
 72 ST- 51
 73 "   IHRE
ZAHL "
 74 "├IST WA
HR-"
 75 PRA
 76 "       SCHE
INLICH "
 77 "├DIE "
 78 ARCL 52
 79 PRA
 80 "        AN
"
 81 ARCL 51
 82 "├.STELL
E"
 83 PRA
 84 STOP
 85♦LBL 01
 86 PRA
 87 ISG 51
 88 GTO 02
 89 .END.
-------------
```

```
                PRP "MAGIC"

 01♦LBL "MAGIC"
FIX 0  CF 29  SF 12  ADV
"UHRZEIT ,SMH?"  PROMPT
STO 00  1.05  STO 51
"MAGISCHE"  PRA
"     ZAHLEN"  PRA  ADV
CF 12

 17♦LBL 02
1.01  STO 52  " "

 21♦LBL 00
RCL 00  9821  *  .211327
+  FRC  STO 00  10  *
INT  X=0?  GTO 00
STO IND 51  ARCL X  "├ "
ISG 52  SF 00  FC?C 00
XEQ 01  ISG 51  GTO 00
ADV  "SUCHEN SIE IHRE"
PRA  "SCHLUESSELZAHL "
"├X AN DER"  PRA
"Y. STELLE."  PRA
"WENN SIE "
"├FERTIG SIND"  PRA
"DRUECKEN SIE "  "├RUN."
PRA  ADV  ASHF  ASHF
PROMPT  1  STO 51

 63♦LBL 05
RCL IND 51  STO 52
ST+ 51  50  RCL 51
X<=Y?  GTO 05  RCL Z
ST- 51  " IHRE ZAHL "
"├IST WAHR-"  PRA
"  SCHEINLICH "
"├DIE "  ARCL 52  PRA
"     AN "  ARCL 51
"├.STELLE"  PRA  STOP

 85♦LBL 01
PRA  ISG 51  GTO 02
.END.
-------------
                    CAT 1
LBL'MAGIC
.END.       336 BYTES
```

BEISPIEL:

```
EINGABE:: ,258654

MAGISCHE
      ZAHLEN

 4 8 1 3 2 7 8 7 2 7
 5 9 5 4 5 5 2 1 9 4
 9 5 7 7 5 4 9 3 8 3
 1 7 2 6 3 3 1 3 6 7
 6 2 1 5 4 2 6 9 4 4

SUCHEN SIE IHRE
SCHLUESSELZAHL X AN DER
Y. STELLE.
WENN SIE FERTIG SIND
DRUECKEN SIE RUN.

 IHRE ZAHL IST WAHR-
  SCHEINLICH DIE 4
   AN 49.STELLE
```

```
EINGABE: ,456789

MAGISCHE
      ZAHLEN

 3 8 7 4 5 1 6 6 2 7
 9 1 7 3 7 7 4 3 7 9
 8 5 9 8 7 8 9 8 5 5
 4 4 4 1 1 7 8 9 9 3
 1 5 4 2 3 7 5 4 8 8

SUCHEN SIE IHRE
SCHLUESSELZAHL X AN DER
Y. STELLE.
WENN SIE FERTIG SIND
DRUECKEN SIE RUN.

 IHRE ZAHL IST WAHR-
  SCHEINLICH DIE 5
   AN 47.STELLE
```

Magische Zahlen

Das Programm beruht auf einer Entdeckung des Physikers Martin D. Kruskal. Magier benutzen diese Entdeckung für Kartentricks. Unter Ihnen ist sie allgemein als Kruskals Prinzip bekannt.

Spielablauf:

Der Spieler stabtet nun das Programm mit XEQ MAGIC und gibt die Uhrzeit (,SSMMHH), oder eine Anfangszahl (zwischen ∅ und 1) ein. Der Rechner erzeugt dann 5o Zufallszahlen von 1....9 . Der Spieler denkt sich eine Zahl, die Schlüsselzahl zwischen 1 und 1o.
Angenommen der Spieler hat sich als Schlüsselzahl die 4 ausgedacht. Die vierte Zahl des Rechnerausdrucks (s.o. 1.Beisp.) ist eine 3, die neue Schlüsselzahl. Er zählt nun drei Zahlen weiter Diese Schlüsselzahl ist die 8. Nach diesem Prinzip zählt er alle 5o Zahlen durch. Die letzte Schlüsselzahl wird nicht zu Ende gezählt werden können. Diese, durch die zufallsabhängige Zählweise erreichte Zahl und die Stelle, auf welcher sie steht, merkt sich der Spieler.
Nun ist der Rechner wieder an der Reihe. Durch drücken von RUN errät er wahrscheinlich die richtige Zahl und die richtige Stelle. Die Wahrscheinlichkeit, daß der Rechner sie errät, liegt bei ca. 5/6, schwankt jedoch von Spiel zu Spiel.

ROBERT KLAUC

EINSER


```
SIZE: 005
START : XEQ 1
EINGABE: ANFANGSZAHL
```

SYSTEM-KONFIGURATION:
KEINE

```
01♦LBL "1"
02 FIX 0
03 CF 29
04 0
05 STO 01
06 STO 02
07♦LBL 03
08 1.002
09 STO 03

10♦LBL 04
11 0
12 STO 04
13 CF 01
14 CF 02
15 SF IND 0
            3

16♦LBL A
17 RCL 00
18 9821
19 *
20 .211327
21 +
22 FRC
23 STO 00
24 6
25 *
26 INT
27 X=0?
28 6
29 "WURF: "
30 ARCL X
31 ST+ 04
32 1
33 -
34 X=0?
35 STO 04
36 "├ Σ="
37 ARCL 04
38 AVIEW
39 X≠0?
40 STOP
41 PSE
42♦LBL B
43 RCL 04
44 ST+ IND
          03
45 RCL IND
          03
46 CLA
47 ARCL X
48 "├ PUNKT
          E"
49 AVIEW
50 PSE
51 101
52 X<=Y?
53 STOP
54 ISG 03
55 GTO 04
56 GTO 03
57 END
```

```
                    CAT 1
LBL'1
END          112 BYTES
.END.         07 BYTES
```

```
 01♦LBL "1"
FIX 0  CF 29  0  STO 01
STO 02

 07♦LBL 03
1.002  STO 03

 10♦LBL 04
0  STO 04  CF 01  CF 02
SF IND 03

 16♦LBL A
RCL 00  9821  *  .211327
+  FRC  STO 00  6  *
INT  X=0?  6  "WURF: "
ARCL X  ST+ 04  1  -
X=0?  STO 04  "├ Σ="
ARCL 04  AVIEW  X≠0?
STOP  PSE

 42♦LBL B
RCL 04  ST+ IND 03
RCL IND 03  CLA  ARCL X
"├ PUNKTE"  AVIEW  PSE
101  X<=Y?  STOP  ISG 03
GTO 04  GTO 03  END
```

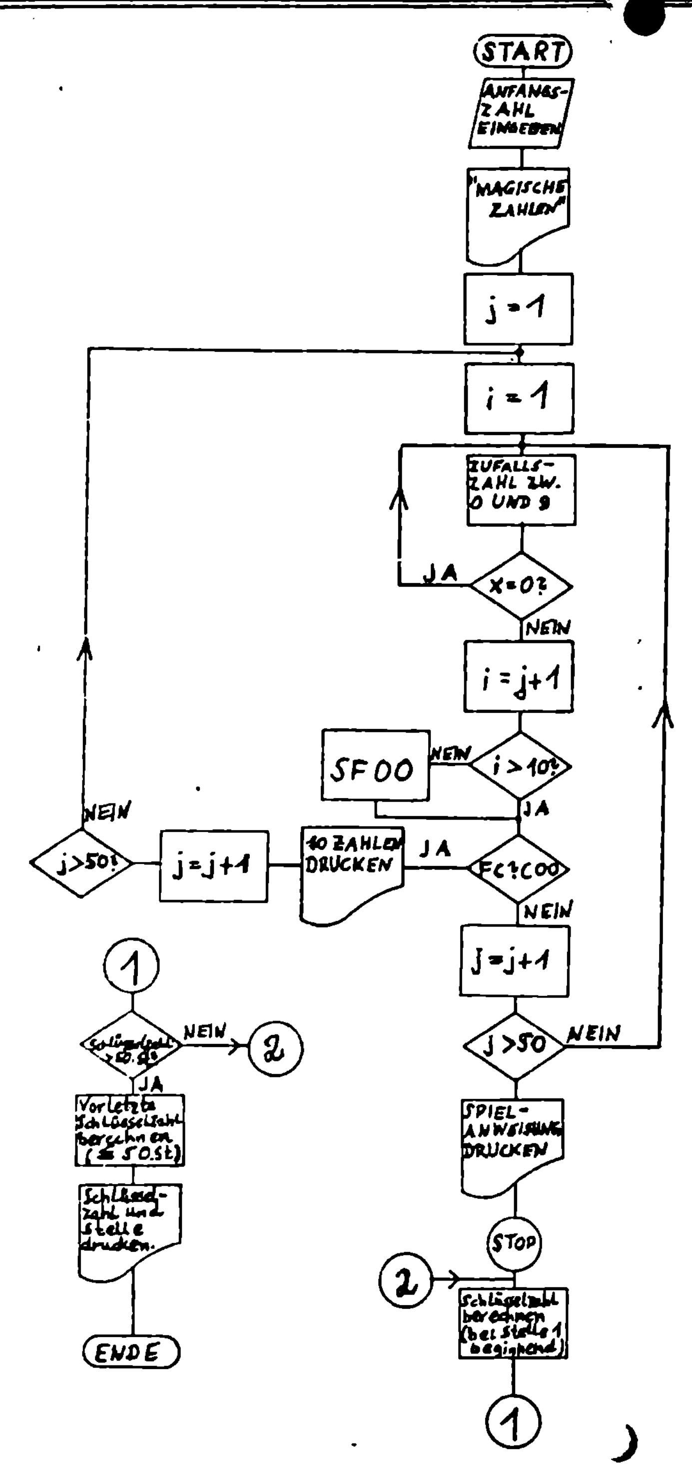
Dies ist ein Spielprogramm für 2 Spieler. Die Würfelwerte werden summiert und gutgeschrieben. Jeder Spieler kann sooft würfeln wie er will. Wird jedoch eine Eins gewürfelt, so ist die Gutschrift Ø und der nächste Spieler ist an der Reihe.
Sieger ist der Spieler, welcher zuerst 1o1 Punkte erreicht.

Begonnen wird das Spiel dadurch, daß in Register oo eine Anfangszahl eingegeben wird . Mit XEQ 1 wird das eigentliche Spiel begonnen. Es beginnt Spieler 1 sofort mit dem ersten Wurf. Der Spieler, der gerade würfelt, wird durch die Flags 1 und 2 symbolisiert. Angezeigt wird der Wurf und die Gesamtpunkte der jeweiligen Runde. Will der Spieler weiterwürfeln, drückt er XEQ A, will er passen, drückt er XEQ B. Angezeigt wird dann die Gesamtpunktzahl des Spielers und der nächste Spieler beginnt automatisch zu würfeln. Ebenso, wenn ein Spieler eine Eins würfelt.

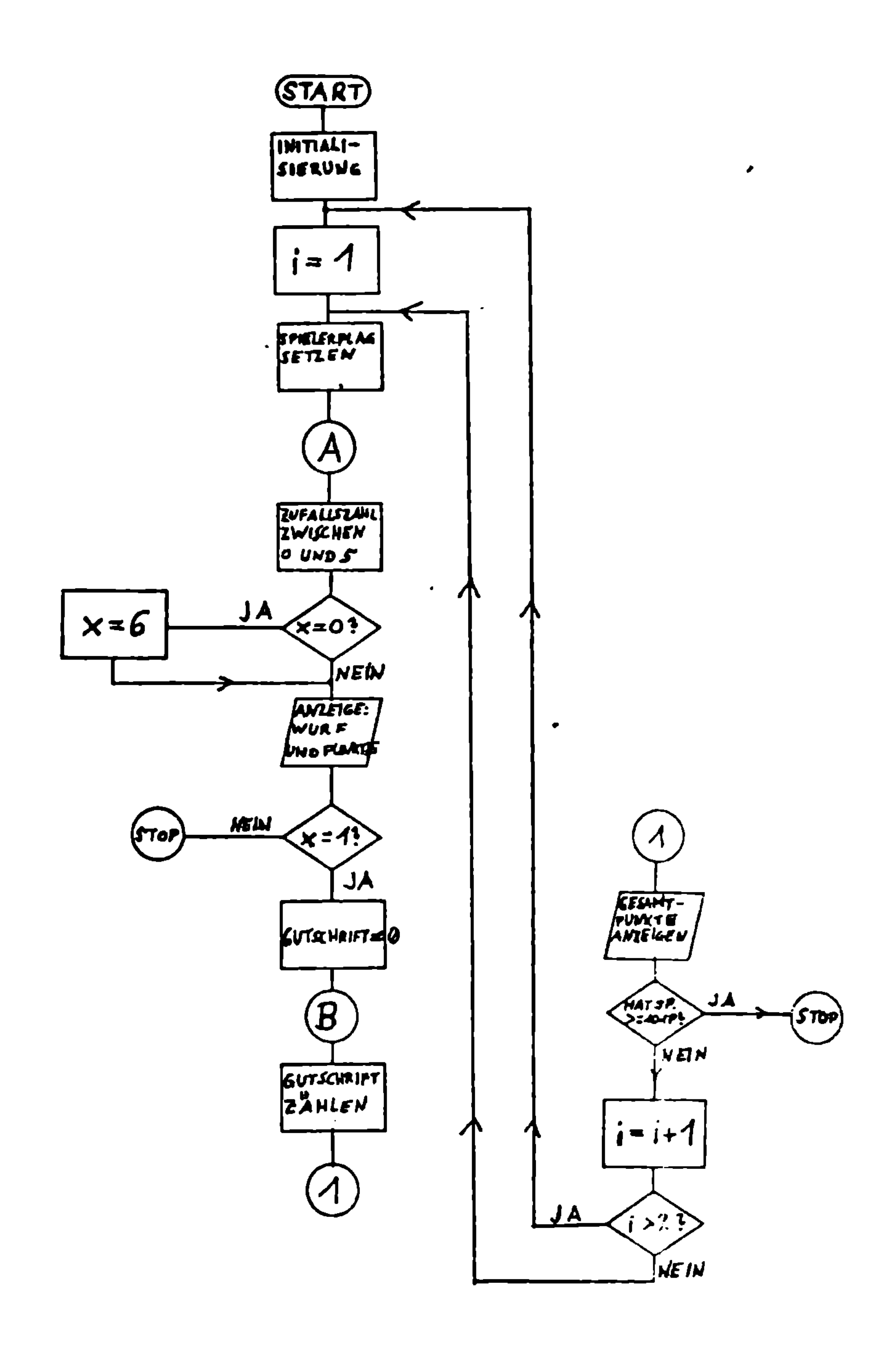
Sobald ein Spieler 1o1 Punkte oder mehr erreicht hat, stoppt der Rechner und am Flag (1 od. 2) ist der Gewinner zu ersehen. Nun bleibt nur noch, Euch viel Spaß zu wünschen und zu hoffen, daß Du eine gute Strategie findest.

ROBERT KLAUC

FLOW-CHART FÜR 'MAGISCHE ZAHLEN'



FLOW-CHART FÜR EINSER



MULTIPLOTTER

PROGRAMM DIENT ZUM PLOTTEN VON BIS ZU 4 FUNKTIONEN GLEICHZEITIG. ES KOENNEN AUCH GROESSERE AUSDRUCKE DURCH NEBENEINANDERLEGEN VON FUNKTIONSABSCHNITTEN ERSTELLT WERDEN.

SIZE: 026
START: XEQ MULTI
RESTART: XEQ NEW
EINGABE: ANZ. D. FUNKT., FUNKTIONSNAMEN, GRENZEN, ACHSE, INKREMENT (BEI NEGATIVER ZAHL WIE BEI PRINTER STANDARTPROGR.).

SYSTEM-KONFIGURATION
1 MEMORY MODULE
PRINTER

```
PRP "MULTI"

01◆LBL "MUL
         TI"
02 FIX 0
03 CF 29
04 4
05 "ZAHL D.
    F(X)?"
06 PROMPT
07 ABS
08 X>Y?
09 GTO "MUL
         TI"
10 1 E3
11 /
12 1
13 +
14 STO 00
15 AON

16◆LBL 00
17 "NAME F"
18 ARCL X
19 "⊢ ?"
20 PROMPT
21 ASTO IND
         X
22 ISG X
23 GTO 00
24 AOFF

25◆LBL "NEW
         "
26 "Y MIN ?
         "
27 PROMPT
28 STO 05
29 "Y MAX ?
         "
30 PROMPT
31 STO 06
32 RCL 05
33 X>Y?
34 GTO "NEW
         "
35 "AXIS ?"
36 PROMPT
37 STO 25

38◆LBL 14
39 "X MIN ?
         "
40 PROMPT
41 STO 07
42 "X MAX ?
         "
43 PROMPT
44 STO 08
45 RCL 07
46 X>Y?
47 GTO 14
48 "X INC ?
         "
49 PROMPT
50 X<0?
51 XEQ 15
52 STO 09
53 CF 12
54 42
55 STO 11
56 10
57 STO 12
58 43
59 STO 13
60 1
61 STO 14
62 RCL 00
63 STO 10
64 ADV

65◆LBL 01
66 " PLOT O
      F: "
67 ARCL IND
         10
68 "⊢   "
69 ACA
70 40
71 ACCHR
72 RCL 10
73 10
74 +
75 RCL IND
         X
76 ACCHR
77 41
78 ACCHR
79 PRBUF
80 ISG 10
81 GTO 01
82 FIX 1
83 " Y: "
84 ARCL 05
85 "⊢ TO "
86 ARCL 06
87 PRA
88 " X: "
89 ARCL 07
90 "⊢ TO "
91 ARCL 08
92 PRA
93 FIX 3
94 " AXIS:
         "
95 ARCL 25
96 PRA
97 " STEPS
      OF "
98 ARCL 09
99 PRA
100 XEQ 13
101 "-------"
102 ASTO L
103 ARCL L
104 ARCL L
105 ARCL L
106 PRA
107 FIX 0
108 RCL 06
109 RCL 05
110 -
111 161
112 /
113 STO 23
114 RCL 05
115 CHS
116 RCL 25
117 +
118 RCL 23
119 /
120 RND
121 X<0?
122 200
123 STO 16
124 .124
125 +
126 STO 24
127 RCL 07
128 STO 15

129◆LBL 02
130 RCL 24
131 STO 16
132 RCL 00
133 STO 10

134◆LBL 03
135 RCL IND
         10
136 STO 21
137 RCL 15
138 XEQ IND
         21
139 RCL 05
140 -
141 RCL 23
142 /
143 RND
144 X<0?
145 200
146 RCL 10
147 10
148 +
149 X<>Y
150 RCL IND
         Y
151 1 E3
152 /
153 +
154 RCL 10
155 16
156 +
157 X<>Y
158 STO IND
         Y
159 ISG 10
160 GTO 03
161 RCL 00
162 16.016
163 +
164 STO 21

165◆LBL 04
166 RCL 21
167 STO 22
168 1
169 -
170 RCL IND
         X

171◆LBL 05
172 RCL IND
         22
173 X<Y?
174 X<>Y
175 STO IND
         22
176 RDN
177 ISG 22
178 GTO 05
179 STO IND
         Y
180 ISG 21
181 GTO 04
182 RCL 16
183 FRC
184 1 E3
165 *
```

```
186 STO 22
187 124
188 X=Y?
189 XEQ 11
190 161
191 RCL 16
192 X>Y?
193 GTO 12
194 SKPCOL
195 RCL 22
196 ACCHR

197•LBL 09
198 RCL 00
199 16.016
200 +
201 STO 21

202•LBL 06
203 RCL IND
          21
204 FRC
205 1 E3
206 *
207 STO 22
208 124
209 X=Y?
210 XEQ 10
211 161
212 RCL IND
          21
213 INT
214 X>Y?
215 GTO 12
216 RCL 21
217 1
218 -
219 X<>Y
220 RCL IND
           Y
221 INT
222 -
223 7
224 FS?C 05
225 CLX
226 -
227 X<0?
228 GTO 08
229 SKPCOL
230 RCL 22
231 ACCHR

232•LBL 07
233 ISG 21
234 GTO 06

235•LBL 12
236 PRBUF
237 RCL 09
238 ST+ 15
239 RCL 08
240 RCL 15
241 X<=Y?
242 GTO 02
243 FIX 4
244 XEQ 13
245 BEEP
246 "  END OF
           "
247 "├MULTIP
    LOTTING"
248 PRA
249 STOP

250•LBL 10
251 RCL 21
252 INT
253 LASTX
254 FRC
255 1 E3
256 *
257 X=Y?
258 RTN
259 1
260 ST+ Z
261 RCL IND
           Z
262 RCL IND
          21
263 -
264 7
265 -
266 0
267 X<=Y?
268 RTN

269•LBL 08
270 RCL 21
271 1
272 -
273 RCL IND
           X
274 STO IND
          21
275 GTO 07

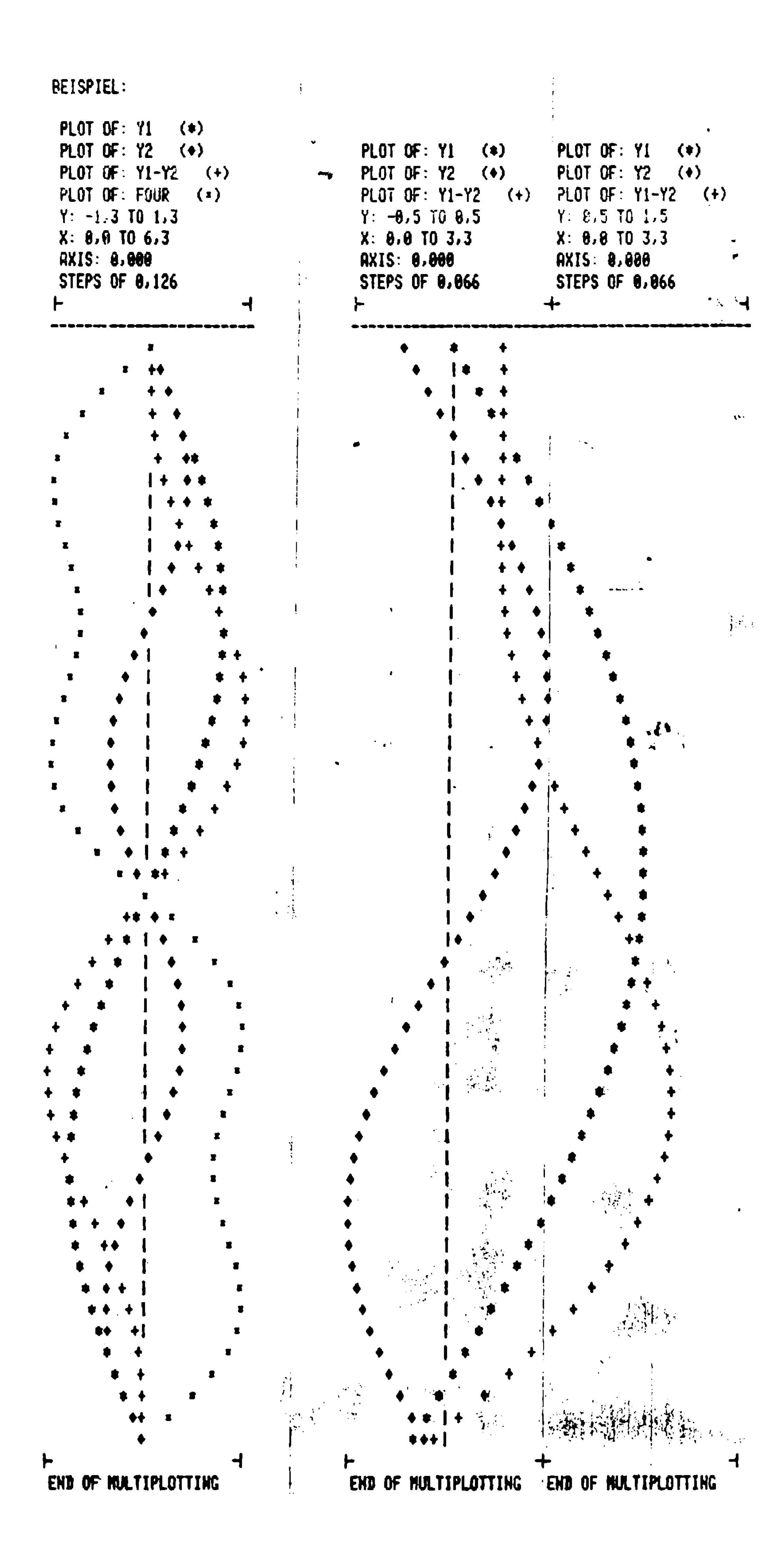
276•LBL 11
277 RCL 17
278 RCL 16
279 -
280 7
281 -
282 0
283 X<=Y?
284 RTN
285 STO 16
286 SF 05
287 GTO 09

288•LBL 15
289 ABS
290 RCL 08
291 RCL 07
292 -
293 X<>Y
294 /
295 RTN

296•LBL 13
297 3
298 SKPCOL
299 127
300 ACCOL
301 8
302 ACCOL
303 ACCOL
304 ACCOL
305 ACCOL
306 ACCOL
307 ACCOL
308 148
309 SKPCOL
310 8
311 ACCOL
312 ACCOL
313 ACCOL
314 ACCOL
315 ACCOL
316 ACCOL
317 127
318 ACCOL
319 PRBUF
320 END
```

```
------------
        CAT 1
        CAT 1
LBL'MULTI
LBL'NEW
END       666 BYTES
.END.     06 BYTES
------------
```

ROBERT KLAUC


BEISPIEL:
PLOT OF: Y1 (*)
PLOT OF: Y2 (♦)
PLOT OF: Y1-Y2 (+)
PLOT OF: FOUR (¤)
Y: -1,3 TO 1,3
X: 0,0 TO 6,3
AXIS: 0,000
STEPS OF 0,126
END OF MULTIPLOTTING
PLOT OF: Y1 (*)
PLOT OF: Y2 (♦)
PLOT OF: Y1-Y2 (+)
Y: -0,5 TO 0,5
X: 0,0 TO 3,3
AXIS: 0,000
STEPS OF 0,066
END OF MULTIPLOTTING
PLOT OF: Y1 (*)
PLOT OF: Y2 (♦)
PLOT OF: Y1-Y2 (+)
Y: 0,5 TO 1,5
X: 0,0 TO 3,3
AXIS: 0,000
STEPS OF 0,066
END OF MULTIPLOTTING

Setzen und löschen aller Flags

Das Programm FLAG dient zum gezielten setzen und löschen einzelner Flags mit der Nummer 30 bis 55. Das Programm wird mit XEQ"FLAG" gestartet, wobei die Nummer des betreffenden Flags in Stack X steht. Ist die Zahl negativ, wird das Flag gelöscht, andernfalls gesetzt. FLAG ist als Unterprogramm ausgelegt, welches jedoch die Speicher Roo bis Ro4 benutzt. Es ist nun möglich zwei neue Indikatorflags (BAT und SHIFT) zu verwenden, sowie Flag 55 zu manipulieren. Um die Auswirkungen zu demonstrieren, sollten FLAG und die folgenden drei Programme eingegeben werden.

```
01◆LBL "1"
02 1000
03 STO 01
04◆LBL 01
05 DSE 01
06 GTO 01
07 TONE 7
08 END
```

```
01◆LBL "2"
02 -55
03 XEQ "FLA
G"
04 1000
05 STO 01
06◆LBL 01
07 DSE 01
08 GTO 01
09 55
10 XEQ "FLA
G"
11 TONE 7
12 END
```

```
01◆LBL "3"
02 0
03 STO d
04 1000
05 STO 01
06◆LBL 01
07 DSE 01
08 GTO 01
09 10
10 STO d
11 SF 26
12 TONE 7
13 END
```

Bei angeschlossenem und eingeschaltetem Drucker kommt man zu folgenden Laufzeiten: "1" 2,15 Min.; "2" 1,55 Min.; "3" 1,39 Min.
Nach Ablauf von "1" und "2" bleiben alle Flags unverändert. Bis auf 55 und 26 werden bei "3" jedoch alle Flags gelöscht. Man kann also mit CF 55 eine nicht unerhebliche Geschwindigkeitssteigerung erzielen. Lustig ist auch 52 XEQ"FLAG". Jedoch sollte man FLAG vorher auf Karte sichern!

Rolf Mach
Thorwaldsenstraße 3
6090 Rüsselsheim

```
01◆LBL "FLAG"
02 STO 00
03 ABS
04 24
05 X>Y?
06 SF IND Y
07 X>Y?
08 RTN
09 RDN
10 56
11 X<=Y?
12 RTN
13 FC? 00
14 .1
15 FC? 00
16 ST+ 00
17 RCL 00
18 X<0?
19 XEQ 01
20 SF 00
21 FC? 48
22 .2
23 FC? 48
24 ST+ 00
25 RCL 00
26 X<0?
27 XEQ 02
28 AON
29 RCL d
30 CLA
31 STO [
32 "├***"
33 RCL [
34 RCL \
35 CLA
36 ASTO 03
37 STO [
38 ASTO 01
39 X<>Y
40 CLA
41 STO [
42 RCL 00
43 8
44 /
45 GTO IND X
46◆LBL 01
47 .2
48 ST- 00
49 RTN
50◆LBL 02
51 .4
52 ST- 00
53 RTN
54◆LBL 06
55 XEQ 00
56◆LBL 05
57 XEQ 00
58◆LBL 04
59 XEQ 00
60◆LBL 03
61 XEQ 00
62 SF 08
63 RCL d
64 STO [
65 "├*"
66 RCL [
67 CLA
68 STO [
69 ASTO 03
70 CLA
71 ARCL 02
72 ARCL 03
73 RCL [
74 STO d
75 RCL 00
76 INT
77 X<0?
78 SF 09
79 ABS
80 8
81 MOD
82 FC? 09
83 SF IND X
84 FS? 09
85 CF IND X
86 RCL d
87 STO [
88 ASTO 02
89 CLA
90 "*****"
91 ARCL 02
92 ASTO 02
93 "*"
94 ARCL 02
95 ASHF
96 ASTO 02
97 CLA
98 ARCL 01
99 ARCL 02
100 ARCL 04
101 ASTO 02
102 ASHF
103 ASTO 01
104 CLA
105 "*****"
106 ARCL 01
107 ASTO 01
108 "*"
109 ARCL 01
110 ASHF
111 ASTO 01
112 CLA
113 ARCL 02
114 ARCL 01
115 RCL [
116 STO d
117 RCL 00
118 ABS
119 FRC
120 X=0?
121 RTN
122 .2
123 X<Y?
124 CF 00
125 RDN
126 .1
127 X<Y?
128 AOFF
129 X<>Y
130 X<=Y?
131 CF 00
132 RTN
133◆LBL 00
134 "├*"
135 RCL \
136 RCL [
137 CLA
138 STO [
139 ASTO 04
140 CLA
141 X<>Y
142 STO [
143 ASTO 02
144 CLA
145 ARCL 01
146 ARCL 03
147 ASTO 01
148 CLA
149 ARCL 02
150 ASTO 03
151 CLA
152 ARCL 04
153 "├*"
154 END
END
273 BYTES
```

Kleiner Testbericht ______________ HP-82143A ______________ Thermodrucker 41c

Für den HP-41c werden Thermodrucker von versch. Herstellern angeboten. In Deutschland habe ich bisher nur den HP-82143A angetroffen, der von HP hergestellt und vertrieben wird. Doch soweit ich gehört und gelesen habe, darf dieser Drucker bisher als am geeignetesten angesehen werden. Daher hier ein T E S T , der keinen Anspruch auf Vollständigkeit und absolute Objektivität erhebt:

Eine der ersten Fragen richtet sich fast immer nach dem Preis. Er liegt für den HP-82143A derzeit bei fast 9oo,--DM. Damit ist der Drucker - wie bei fast allen kleineren Computersystemen - die teuerste Einheit. Warum sich diese Ausgabe für den Anwender, der nicht nur ein zeitweiliges Spielzeug sucht, lohnt - oder auch nicht - sei im folgenden analysiert:

Der Drucker vermittelt ein mechanisch stabilen Eindruck und tut sich durch klare Gliederung der Funktionstasten hervor. Während der viermonatigen Testzeit stellte sich kein mechanischer Defekt ein (auch kein sonstiger), doch gab ein seit Kauf vorhandener Wackelkontakt an der Ladebuchse Anlaß zum Ärgern. Außerdem fiel negativ auf, daß dem Drucker keine Schutztasche beiliegt. Selbst wenn der Drucker nur selten mobil gebraucht wird, so müßte eine Schutztasche, wie sie auch dem HP-41C beiliegt, als Selbstverständlichkeit gelten.
Es wird Thermopapier verwendet, was die bekannten Vorteile von scharfem Schriftbild und fehlendem Farbband bringt. Doch gleichzeitig muß darauf hingewiesen werden, daß Thermopapier - besonders blau druckendes - sehr leicht verbleicht und zudem nicht gerade billig ist.
Im Zusammenhang mit seiner 7x9 Matrix weist der Drucker ein für Matrixdrucker schönes Schriftbild auf, da horizontal nebeneinanderliegende Punkte zu einem Strich verbunden werden (genaugenommen also keine "Punkte", sondern "Quadrate"). Neben den 128 verschiedenen Zeichen, die zu den Standardzeichen des HP-82143A gehören (7x7 Matrix), läßt sich jedes andere durch eine 9x7 Matrix darstellbare Zeichen erzeugen und abspeichern, wenngleich deutlich mehr Speicherplatz benötigt wird.
Der Drucker verfügt über zahlreiche Druckfunktionen, die dem Anwender sehr vielseitige Möglichkeiten eröffnen, sowie ein 34o Zeilen langes Programm im Festspeicher (ROM), das zum Plotten einer programmierten Funktion mit Skalierung dient. Eine leistungsfähigere Plottroutine erscheint parallel zu diesem Test. Mit ihr können bis zu vier Graphen gleichzeitig - auch in mehrfacher Papierbreite - geplottet werden. Den eingehenden Test aller einzelnen Funktionen zu veröffentlichen, würde sicherlich den Rahmen dieses "Kleinen Testberichts" sprengen, weshalb gleichzeitig mit diesem Test die Kurzanleitung des HP-82143A mit der freundlichen Genehmigung von Hewlett-Packard GmbH, Frankfurt/M. erscheint. Als Besonderheit ist aber noch ein bis zu max. 43 Zeichen speichernder Druckbuffer (Zwischenspeicher) zu erwähnen.
Unter Hinweis auf die sehr reichhaltigen Funktionen des HP-82143A, sowie dessen versch. Schriftarten (groß und klein , breit und eng) möchte ich diesen Testbericht mit einer tabellarischen Zusammenfassung beenden:

++++++++++++++++++++

- 27 z.T. erstklassige Druckfunktionen
- 3 versch. Druckermodi (MAN,TRACE,NORM)
- Druckbuffer mit max 43 Zeichen
- Plottprogramm im ROM
- schönes Schriftbild
- 128 ASCII-Zeichen standardmäßig
- eingebauter Akku
- kleines Gehäuse
- Gutes Bedienungshandbuch
- bekannt guter HP-Service im Garantiefall

- Walze weder soft- noch hardwaremäßig steuerbar
- Papier verbleicht (allg. bei Termop.)
- keine Schutztasche
- relativ teuer

Oliver

Thermodrucker HP 82143A Kurzanleitung

Wenn der Thermodrucker HP 82143A an den HP-41C angeschlossen ist, werden die nachfolgenden Operationen innerhalb des Systems wirksam. Diese Operationen können ebenso wie Programme, die diese Operationen beinhalten, nur ausgeführt werden, während der Drucker angeschlossen und eingeschaltet ist. Mit [CATALOG] 2 können Sie diese Funktionen wie auch die übrigen Funktionen angeschlossener Systemerweiterungen auflisten. In den Schalterstellungen MAN und NORM werden die Funktionen lediglich angezeigt, wogegen Sie in der Betriebsart TRACE auch ausgedruckt werden.

Verzeichnis der Funktionen

- [ACA] Akkumuliert ALPHA-Register im Druck-Buffer
- [ACCHR] Akkumuliert Zeichen im Druck-Buffer. Speichern Sie die entsprechende Zahl (1–127) in X und führen Sie [ACCHR] aus.
- [ACCOL] Akkumuliert Spalte im Druck-Buffer. Speichern Sie die Spalten-Druck-Zahl (0–127) in X und führen Sie [ACCOL] aus.
- [ACSPEC] Akkumuliert ein mit [BLDSPEC] In X gebildetes Sonderzeichen im Druck-Buffer.
- [ACX] Akkumuliert X-Register im Druck-Buffer.
- [ADV] Rückt das Papier um eine Zeile vor und druckt den Bufferinhalt rechtsbündig aus.
- [BLDSPEC] Baut ein Sonderzeichen auf. Speichern Sie die Spalten-Druck-Zahl (0–127) in X und führen Sie [BLDSPEC] aus. Ist bis zu siebenmal zu wiederholen. Ergebnis ist mit [ACSPEC] im Druck-Buffer zu akkumulieren oder in ein Register zu speichern.
- [LIST] Listet Programmzeilen auf. Speichern Sie die entsprechende Zahl in X und führen Sie [LIST] aus. Nicht programmierbar.
- [PRA] Druckt ALPHA-Register.
- [PRAXIS] Druckt Achsen aufgrund folgender Eingabewerte: R_{00} = YMIN, R_{01} = YMAX R_{02} = nnn, R_{04} = AXIS
- [PRBUF] Druckt den Inhalt des Buffers linksbündig aus.
- [PRFLAGS] Druckt Flags und weitere Status-Informationen.
- [PRKEYS] Druckt geänderte Tastenbelegungen.
- [PRP] Listet das gewünschte Programm auf. Führen Sie [PRP] aus und tasten Sie den Namen ein. Wenn kein Name angegeben wird, wird das augenblickliche Programm aufgelistet (nicht programmierbar).
- [PRPLOT] Plottet die mit Namen angegebene Funktion. Verlangt folgende Eingabewerte: Name, YMIN, YMAX, AXIS, XMIN, XMAX, XINC. Positive XINC-Werte geben das X-Inkrement an, negative XINC-Werte die Anzahl der gewünschten Inkremente. Auf Wunsch kann ein besonderes Plot-Zeichen in R_{03} gespeichert werden.
- [PRPLOTP] Druckt den Plot der Funktion aus. Ausgangsdaten müssen in folgenden Registern gespeichert sein: R_{00} = YMIN, R_{01} = YMAX, R_{03} = Plot-Sonderzeichen (falls gewünscht), R_{04} = AXIS, R_{08} = XMIN, R_{09} = XMAX, R_{10} = XINC, R_{11} = NAME.
- [PRREG] Druckt die Inhalte aller Register ab R_{00}.
- [PRREGX] Druckt Registerinhalte, durch X gesteuert. Das Format von X ist bbb.eee, wobei bbb das erste und eee das letzte zu druckende Register angibt.
- [PRΣ] Druckt Inhalte der Statistik-Register.
- [PRST] Druckt Stack-Register-Inhalte.
- [PRX] Druckt X-Registerinhalt.
- [REGPLOT] Plottet Einzelzeile unter Verwendung der Daten in folgenden Registern: X = Plot-Wert, R_{00} = YMIN, R_{01} = YMAX, R_{02} = nnn.aaa, wobei nnn die Plotbreite in Spalten und aaa die Spaltenposition der Achse angibt. [PRAXIS] speichert nnn.aaa automatisch nach R_{02}.
- [SKPCHR] Überspringt Zeichen beim Akkumulieren im Druck-Buffer. Speichern Sie die Anzahl der zu überspringenden Zeichen in X und führen [SKPCHR] aus.
- [SKPCOL] Überspringt Spalten bei der Akkumulation im Druck-Buffer. Speichern Sie die Anzahl der zu überspringenden Spalten in X und führen [SKPCOL] aus.
- [STKPLOT] Plottet Einzelzeile unter Verwendung der Daten in den Stack-Registern: T = Plotwert, Z = YMIN, Y = YMAX, X = nnn.aaa, wobei nnn die Plotbreite in Spalten und aaa die Spaltenposition der Achse angibt.

Drucker-Wahlschalter

In Stellung MAN erfolgt Ausdruck nur, wenn Druck-Funktionen von Hand oder im Rahmen eines Programms ausgeführt werden. In der Betriebsart NORM werden Zahlen und ALPHA-Ketten, die eingetastet werden, Funktionsnamen bei der Ausführung über das Tastenfeld und die Ausgabe von Druck-Funktionen gedruckt. In der Stellung TRACE werden Zahlen und ALPHA-Ketten gedruckt, die Sie eintasten, Funktionsnamen, Zwischen- und Endergebnisse sowie die Ausgabe von Druck-Funktionen.

[PRINT]-Taste

Im HP-41C Normal-Modus bewirkt [PRINT] den Ausdruck des Inhaltes des X-Registers. Im ALPHA-Modus druckt [PRINT] den Inhalt des ALPHA-Registers aus. Im PRGM-Modus bewirkt [PRINT] die Einfügung einer [PRX]-Anweisung als Zeile in den Programmspeicher; befindet sich der HP-41C dabei außerdem im ALPHA-Modus, wird eine [PRA]-Anweisung in das Programm eingefügt.

Papier-Vorschub-Taste

Rückt das Druckpapier vor und druckt den Inhalt des Papiers rechtsbündig aus. Im PRGM-Modus wird eine [ADV]-Anweisung in das Programm eingefügt.

Standard HP-41C Druck-Funktionen

[VIEW] zeigt den Inhalt des gewünschten Speicherregisters an und druckt ihn aus. [AVIEW] zeigt den Inhalt des ALPHA-Registers an und druckt ihn aus. [PROMPT] zeigt den Inhalt des ALPHA-Registers an, druckt ihn aus und unterbricht die Programmausführung.

XROM-Funktionen

Programme, die Druck-Funktionen beinhalten, zeigen bei nicht angeschlossenem Drucker eine XROM-Zahl an. Einzelheiten dazu finden Sie im Bedienungshandbuch des HP 82143A.

Flags und der Drucker

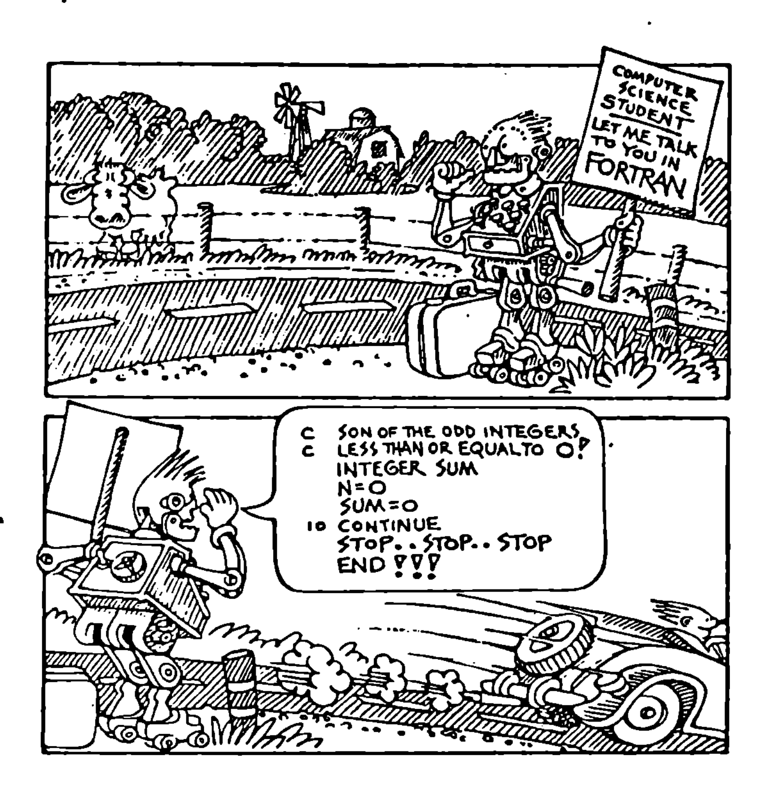
Flag 12: Zeichenbreite-Flag. Bei gesetztem Flag werden alle Zeichen doppelt breit gedruckt.

Flag 13: Kleinschreibungs-Flag. Bei gesetztem Flag werden Kleinbuchstaben gedruckt.

Flag 25: Fehlerignorierungs-Flag. Bei gesetztem Flag wird der erste auftretende Fehler ignoriert. Das Flag wird dabei automatisch gelöscht.

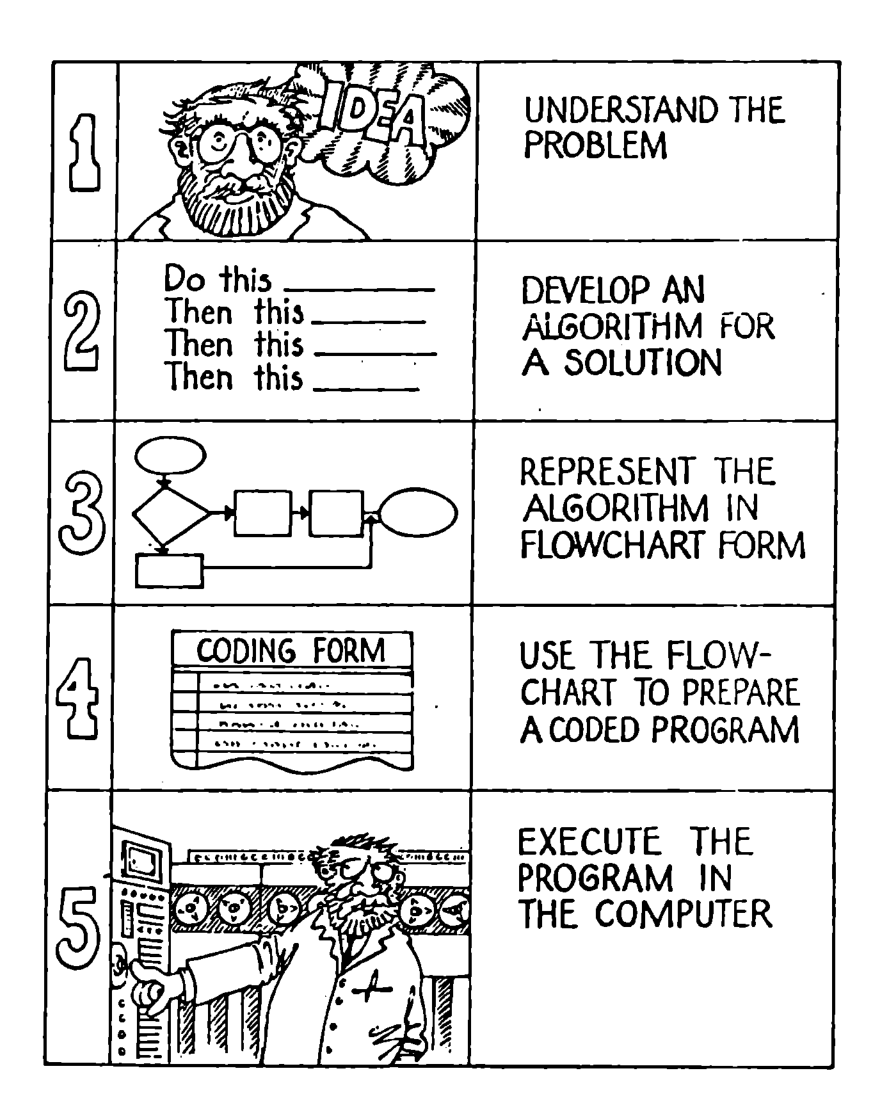
Flag 21: Drucker-Einschalt-Flag. Steuert den Druck bei der Ausführung von Programmen. Siehe Tabelle I.

Flag 55: Drucker-Anwesenheits-Flag. Bestimmt, ob der Drucker angeschlossen ist oder nicht. Siehe Tabelle 1.




```
 01♦LBL "TRA
MP"
AON
 03♦LBL 00
"AUTO?" STO
P  ASTO X
"NO"  ASTO Y
X=Y?
GTO 00  1,00
9  STO 00
CF 23
 14♦LBL 01
"STOP"  AVIE
W
TONE IND 00
PSE
FS?C 23  GTO
 02  ISG 00
GTO 01  "BLE
CH-BONZE"
AVIEW  PSE
TONE 0
"PS-BLOEDEL"
AVIEW  PSE
TONE 1  CLD
GTO 00
 33♦LBL 02
"NACH EUTIN
?"  AVIEW
FND
```

Idee: Detlev Bock


1
IDEA
UNDERSTAND THE PROBLEM
2
Do this
Then this
Then this
Then this
DEVELOP AN ALGORITHM FOR A SOLUTION
3
REPRESENT THE ALGORITHM IN FLOWCHART FORM
4
CODING FORM
USE THE FLOW-CHART TO PREPARE A CODED PROGRAM
5
EXECUTE THE PROGRAM IN THE COMPUTER

Lieber Oliver Rietschel, liebe HP-41C-Fans!

Als ich zum ersten Mal von dem HP-41C Anwenderclub hörte war ich etwas erstaunt, daß es noch mehr Leute gibt, die sich damit befassen, welche Unmöglichkeiten auf dem Rechner möglich sind. Auch ich habe nämlich schon viel auf diesem Gebiet gearbeitet und Erfahrungen gesammelt, die ich gerne weiterzugeben bereit bin. Das meiste davon ist den beigefügten Tabellen, Diagrammen und Kurzbeschreibungen zu entnehmen. Außerdem habe ich ein Instrumentarium entwickelt, welches auf WALL-Karten beiliegt. Diese verwandeln den HP-41C in ein wahres Bastel- und Versuchslabor. Doch zuerst ein wenig Theorie!

Adressen

Wie aus der Tabelle hervorgeht, existieren nicht nur die Befehle STO/RCL usw. M bis Q, sondern viele weitere:

(1)00,(1)01, A bis J

Mit diesen Adressen lassen sich die Datenregister 100 bis 112 direkt ansprechen.

T,Z,Y,X,L

4 Stacks und Last x

M,N,O und die ersten 5 Byte von P

Alpharegister. Die letzten beiden Byte von P werden u.a. als Zählregister bei Durchführung der Catalog-Funktion verwendet sowie bei der Dateneingabe, um z.B. festzustellen, ob eine Ziffer $\neq 0$, das Komma oder der Exponent eingegeben wurde.

Q-Register

Zwischenspeicher, z.B. bei der Dateneingabe, beim Suchen von alphanumerischen Labels (die daher nur 7-stellig sein können!) oder bei ASN.

⊢-Register

Erste 4 1/2 Bytes: Flags für ungeshiftete Tasten. Für jede Taste existiert ein Flag, der bei Belegung derselben gesetzt ist (siehe Diagramm).
Byte ab 4 1/2: Zuletzt durchgeführt Funktion
Letztes Byte: Bei Durchführung von ASN die Taste (siehe Tabelle)

a- und b-Register

Lezte beiden Bytes von b: Zeiger auf die augenblickliche Position im Programmspeicher.
zwei-byte-weise rückwarts von b nach a: Rücksprung=positionen aus Unterprogrammen (genau 6!)

c-Register:

Statusinformationen (siehe Diagramm). Vorsicht: falls 3. und 4. Byte nicht stimmen, dann MEMORY LOST. (warum?)

d-Register

Flags. Siehe Diagramm.

e-Register

erste 4 1/2 Bytes: Flags für geshiftete Tasten
letzte beiden Bytes: Nummer des Programmschrittes, der sich gerade im Speicher befindet (Dieses hat eigentlich keine tiefere Bedeutung und wird meist neu vom HP-41C berechnet)

Zur Datenspeicherung

Aufbau eines Datenregisters:

Erstes Halbbyte:	0	positive Zahl
	1	Alphastring
	9	negative Zahl

Nach Ausführung von ASTO-Befehlen entspricht dem zweiten Halbbyte in dem entsprechenden Register das zweite Halbbyte des 7. Bytes im Alpha-Register (offensichtlich ein Fehler von HP). Die nächsten Bytes enthalten die entsprechenden Stringzeichen.

Bei Zahlen:

Halbbytes 2 - 11	Stellen der Zahl
Halbbyte 12	Vorzeichen des Exponenten (0=+,9=-)
Halbbytes 13 u. 14	Exponent (100-Exp.,falls negativ)

Was entspricht etwa 1,917585 als Programmregister?

0x16+1= 1	LBL 00
9x16+1=145	STO
7x16+5=117	Adresse M
8x16+5=133	RTN
Rest = 0	Leerstellen

Es ist auch möglich, für das Vorzeichen andere Möglichkeiten als 0, 1, und 9, für das Vorzeichen des Exponenten andere als 0 und 9 sowie Ziffern größer als 9 zu erhalten

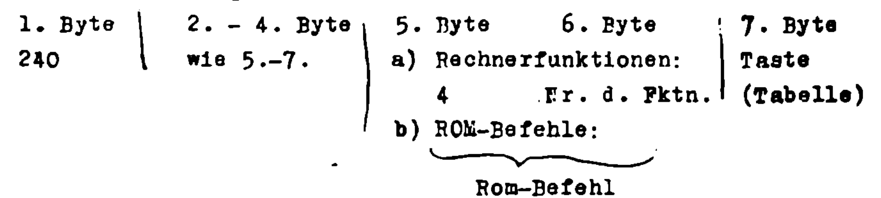
(Aufbau eines Alphastrings der Länge 7, RCL M; siehe unten). Durch Befehle wie STO,RCL,VIEW 00-99, (1)00, (1)01, A-J sowie durch indirekte Adressierung werden diese Daten in normale Zahlen und Strings umgewandelt und z.T. zerstört. Erhalten bleiben sie nur durch Befehle von STO T bis STO e und entsprechend. *

Zur indirekten Adressierung

1 bezeichnet das zweite Datenregister, 0 das erste, soviel ist klar. Darunter aber schließt sich der Kreis (siehe Skizze über Speicheraufbau)! 1023 bezeichnet das erste erste Programmregister, 1022 das zweite etc. Andererseits weiß der Rechner, daß eine vierstellige Adresse unmöglich ist und zeigt NONEXISTENT an. Aber (zumindest bis Seriennummer 1950) mit 999 läßt sich das 24. Programmregister ansprechen, mit 998 das 25. usw. (Vorsicht!Register können dabei verändert werden, siehe Datenspeicherung allgemein!). Am Ende des Programmspeichers und des Key-Assignments folgt eine Lücke von 176 Registern, durch die dann folgenden niedrigeren Adressen lassen sich die 16 Statusregister T - e ansprechen. Diese Adressen kann man aus dem Size und der Anzahl der Memory-Module berechnen.

Aufbau des Key-Assignments

Die entsprechenden Register stehen linksbündig am Anfang des RAM's (siehe Skizze über Speicheraufbau).
Aufbau eines Registers:



Die Bytes 5-7 werden beim Belegen zuerst aufgefüllt, danach 2-4. Zu a) die Nummern der entsprechenden Funktionen sind zumeist der Tabelle zu entnehmen (+ = 64, GTO = 208). Das Erstaunliche: Es gibt genau 256 belegbare Funktionen, die z.T. interessante Eigenschaften haben(z.B. 32 oder 60).

* Die selbe Schwierigkeit tritt auch bei Befehlen des Kartenlesers wie WDTA usw. auf.

Interessant auch die nicht-programmierbaren Funktionen:
CAT=0, DEL=2, COPY=3, CLP=4, SIZE=6, BST=7, SST=8,
ON=9, PACK=10, ASN =15
Nun aber:
1 = GTO. Rechner springt zu dem (echten) Programmschritt
der Nummer, die im e-Register angegeben ist
5 = R/S, 11 = Programmschritt löschen, 12 = USER off, ALPHA
on, je nach belegter Taste, 13= ???, 14 = SHIFT !!!
Der beigefügte Ausdruck gibt die Namen der Funktionen von
0-255 in der richtigen Reihenfolge an (teilweise Unter=
schied zwischen Rechner und Drucker).
ZU b) Statt ROM-Befehlen lassen sich hier auch andere
2-Byte-Befehle belegen, z.B. FIX 8 oder RCL M.

Nach soviel Theorie nun aber zur Praxis,zu den beigefüg-
ten Karten, den Programmen und den Keys darauf.

"BASTEL"

(Offensichtlich nur bis Seriennummer 1950 wirksam)
Diesew Programm erlaubt es, eine Folge von 6 beliebigen
Bytes an eine bestimmte Stelle im Programmspeicher zu
schreiben. Hierbei wird ausgenutzt, daß mit der Adresse
999 das 24. Programmregister angesprochen wird. Deshalb
darf kein anderes Programm vor dem Bastel-Programm im
Programmspeicher stehen! Dabei wird im Flag-Register
ein Programm gebildet, das ein Stringzeichen anhängt,
das Stringzeichen kann durch setzen der Flags 16 bis
23 beliebig gewählt werden (Schritte 36 - 53). Dieses
wird dann in das 24. Programmregister geschrieben und
mit XEQ a aufgerufen (wegen Schritt 39,40 und 41 ist
dabei Flag 31 gesetzt). Auf diese Weise wird ein Alpha-
string der Länge 6 erzeugt, der dann endgültig in das
24. Register eingeschrieben wird, wo der Rechner das
Programm anhält, da nach Ausführung des Programms der
Flag 31 wieder gelöscht ist (alle Flags ab Nr. 24
bleiben erhalten, da das d-Register im a-Register ge-
speichert wird).
Anleitung
Eingabe: 1. Möglichkeit: 1.Byte in R01 eingeben, 2. in
R02 ... 6. in R06 speichern,XEQ "BASTEL" bzw.
XEQ A , c eingeben

2. Möglichkeit: XEQ "BASTEL" oder XEQ A,
1. Byte eingeben, warten, zweites Byte eingeben usw. Zum Abbruch des Programms negative Zahl eingeben.

Schaltet man nach Anhalten des Programms auf PRGM, erscheint zunächst eine 0, dahinter dann die gewünschten Bytes als Programmbefehle. Beispiel: 144,126,146,129,241,38: RCL d, ST+IND 01, "&".

Vorsicht! Man muß aufpassen, daß durch das Basteln nicht unkontrolliert END-Befehle, Alpha-Labels oder gar der .END.-Befehl zerstört werden (deshalb auch der RDN-Puffer am Ende, falls etwa versehentlich als 6. Byte 248 eingegeben wurde, das, wenn es selbst frei ist, die 8 folgenden Bytes auffrißt). Durch XEQ C läßt sich aber der Zustand des Programmregisters vor dem Basteln wiederherstellen (oder manuell 999, RCL09, STO IND Y). Das neu erzeugte Programmregister ist außerdem in R08 gespeichert. Der Inhalt eines beliebigen Programmregisters läßt sich durch Angabe der Nummer und XEQ B in das 24. Programmregister schreiben. Durch XEQ a kann man jederzeit zum 24. Register springen.

Tips: Die 0 am Anfang des Registers sollte stets hier stehenbleiben. Sie bewirkt, daß die 6 folgenden Bytes durch STO IND und RCL IND 999 nicht verändert werden, da sie als Stringzeichen interpretiert und damit akzeptiert werden. Aufbau von Befehlen länger als 6 Bytes: 245 und letzte 5 Bytes eingeben und Basteln. Durch die 245 werden die 5 Bytes in einen Alphastring absorbiert. Dies kann wichtig sein: Wäre das letzte Byte etwa 255, so könnten die nachfolgenden Befehle aufgefressen werden. Nach Anhalten des Programms im 24. Register hinter der 0 fünfmal ENTER↑ eingeben, PACK. Nächste Bastel-Eingabe: neutrale Befehle (131=ENTER), danach die restlichen Bytes des Befehls (oder 250 und die nächsten 5 Bytes des Befehls und entsprechend weiter). Häufige Anwendung: Herstellen eines Alphastrings der Länge 7:

1. Basteln: 245, letzte 5 Bytes des Strings
2. Basteln: 131,131,131,247, erste beiden Bytes

Durch einen Alphastring der Länge 7 gefolgt von RCL M läßt sich im x-Register jede beliebige "Zahl" und jeder String erhalten und dadurch z.B. mit STO d jeder

beliebige Flag-Status! Interessante Anwendungsmöglichkeiten auch für den Drucker: Es kann direkt ein Alphastring für ACSPEC gebildet werden, auch können Alphastrings mit Kleinbuchstaben und anderen Zeichen erzeugt werden, die auf dem Rechner aber meist Unsinn liefern.

"SAVE" (bis Seriennummer 1950)

Falls zum Beispiel durch Basteln der CAT 1 durcheinandergeraten ist, insbesondere der .END.-Befehl gelöscht wurde, kann man den Programmspeicher und die User-Keys retten: Programm starten, vor dem Komma Anzahl der (Status-, nicht Programm-) Keys angeben, hinter dem Komma die Anzahl der im Rechner steckenden Memory-Module,R/S. Im CAT 1 erscheint nun lediglich der .END.-Befehl, nach PACK werden aber auch die anderen END-Befehle und Labels in den Katalog wieder aufgenommen. Nach Ausführung des Programms sind keine Datenregister mehr vorhanden, außerdem ist die Position der Statistik-Register verändert worden.
Schwierigkeit: Das Programm zu finden, wenn der CAT 1 nicht mehr existiert. Lösung: Sich beständig mit SST nach vorn tasten, END-Befehle lassen sich durch GTO.nnn überwinden (nicht löschen!!!) oder mit dem Befehl "Goose", siehe unten.

"CSTR"

Stellt einen beliebigen Alphastring der Länge 2 her.
1. Byte in R01 eingeben, 2. Byte in Ro2, XEQ "CSTR".
String erscheint im Alpha- und im x-Register.
Unterschied zu "BASTEL": nur 2 Byte, dafür schneller, Programmregister nicht verändert, funktioniert auf jedem Rechner.

"CNT"

Zählt im x-Register die letzten beiden Bytes aus und gibt sie als mmm,nnn an.

"JMP" (Greift auf "CSTR" zurück)

Dieses Programm erlaubt es, mit dem Programmzeiger an jede beliebige Stelle im RAM zu springen: Sprungposition angeben, XEQ "JMP", danach SST (STO b).

Angabe der Sprungadresse:
INT= Nummer des Registers (0-15 Statusregister T-e,
ab 192 Key-, Programm- und
Datenregister)
FRC= Nummer des Bytes (0,6= erstes Byte, 0,5 = zweites
Byte usw.)
Der Rechner speichert die Programmposition im b-Register nicht in dieser Form ab, sondern zählt immer ein Byte dazu. Ansonsten steht im ersten Halbbyte der betreffenden beiden Bytes die Nummer des Bytes, in den restlichen 1 1/2 die des Registers. Anders in ROM's: Im ersten Halbbyte steht hier die Position des ROM's, in den restlichen 1 1/2 die Nummer des Bytes (0-4095).

"CTLG" (greift auf "CSTR" zurück)

Dieses Programm erlaubt es, CAT 2 und CAT 3 von jeder beliebigen Stelle aus zu starten. Eingabe: INT=Stelle im Katalog, FRC=0,2 oder 0,3 je nach Katalog. Dann XEQ "CTLG", nach Stop des Programms auf R/S oder SST drücken. Interessant: Im CAT 3 steht ab Stelle 118 teilweise unterbrochen der Katalog für das Key-Assignment (Namen der einzelnen Funktionen).

"NORMAL" u. "ABN"

Bitte selbst testen!

Diese Programme zeigen wohl deutlich, welche ungeahnten Möglichkeiten in dem HP-41C schlummern und was man mit "unmöglichen" Befehlen alles anfangen kann. Doch nun zu den User-Keys auf den WALL-Karten.

"Goose"-Befehl (≻ auf Rechner, . auf Drucker) auf -23

Diese Funktion hat die Nummer 178 und entspricht dem Befehl GTO 01. Sie erlaubt es, im RUN-Modus manuell Sprünge bis zu 15 Registern auszuführen gemäß der anzugebenden Adresse: Die Wirkungsweise dieser Adresse entspricht dem zweiten Byte eines Kurzform-Sprungbefehls (siehe Diagramm). Also:

- . 16 1Byte vorwärts
- . 32 2Bytes vorwärts
- . 01 1Register=7Bytes vorwärts
- . 17 1Register/1Byte=8Bytes vorwärts
- . 02 2Register vorwärts usw.
- . IND Rückwärtssprünge

Der Befehl erlaubt es vor allen Dingen, über END-Befehle hinwegzuspringen und auch über den .END.-Befehl hinweg zu den Key-Assignments und den Statusregistern vorzudringen. Dagegen ist es nicht möglich, vom Programmspeicher aus zurück in die Datenregister zu springen.
Anders ist in ROM's zu verfahren, da hier die Daten anders organisiert sind: . IND 01 für ein Byte vorwärts, . IND 02 für zwei Bytes usw., . 01 für ein Byte rückwarts usw.

STO b und RCL b (auf 23 und 24)
Man darf sich nicht daran stören , daß STO b als XROM 05,60 und RCL b als XROM 01,60 gelesen wird (Vergl. auch Aufbau des Key-Assignments, letzte Bemerkung). Diese beiden Keys ermöglichen es auf recht elegante Weise, jederzeit und überall Befehle wie etwa STO M zu basteln. Wie schon mehrfach erwähnt, enthalten die letzten beiden Bytes des b-Registers die Position des Rechners im Programmspeicher, wenn man will, kann man diese beiden Bytes nach RCL b mit dem "CNT"-Programm analysieren. Man kann nun den Rechner überlisten, wie ein kleines Beispiel demonstrieren möge (Vgl. dabei jeweils mit Tabelle): Man gebe an einer beliebigen Stelle im Speicher zweimal (am Programmanfang besser dreimal) den Befehl ENTER ein und schalte dann sofort auf RUN zurück, wo man RCL b ausführe. Nun lösche man die (letzten) beiden ENTER-Befehle und gebe stattdessen RCL IND Z ein, dahinter RCL IND 17 und RDN. Jetzt schalte man wieder auf RUN um, wo man noch den Inhalt von Register b findet, den man vorher abgefragt hat. Nun der Trick: diesen speichere man mittels STO b wieder im b-Register ab und schalte danach auf PRGM um. Der Rechner befindet sich nun da, wo vorher der zweite (dritte) ENTER-Befehl stand. Hier steht nun die Adresse IND Z, d.h. eine 241. Dies liest der HP-41C jetzt aber als Alphastring der Länge 1, und dafür braucht er ja auch ein Alphazeichen. Als solches liest er nun das erste Byte des nächsten Befehles, also von RCL IND 17. Führt man nun SST aus, so findet der Rechner hinter dem String die verbleibende Adresse IND 17 entsprechend Byte 145, was er natürlich als erstes Byte eines STO-Befehles interpretiert. Deshalb

liest er das nächste Byte nicht als RDN, sondern als Adresse M (117). Dies hat aber keine besondere Bewandtnis, weil sich der Rechner ja praktisch bloß verirrt hat. Wenn man jedoch nach STO b den vorgefundenen Alphastring löscht, so löscht man in Wirklichkeit die Adresse IND Z und das RCL von RCL IND 17. Damit hat man aber die Adresse IND 17 "befreit" und den Befehl STO M erzeugt, im Programmspeicher steht nun RCL 00 (Langform!), Leerstelle, STO M.

Auf analoge Weise lassen sich noch viele andere Befehle konstruieren und man wird sich bald eine gewisse Routine darin aneignen. Noch ein Beispiel: Wir löschen nun RCL 00 und geben stattdessen ein: RCL IND Z, RCL IND 64, ENTER, "AA". Im RUN-Modus führen wir STO b aus (falls das x-Register nicht seit dem letzten Basteln abgeändert wurde), löschen wieder den Alphastring, im Programm steht dann der Befehl ... nicht zu fassen,wie? Nach Pack steht er dann auch im CAT 1 (bevor dies nicht der Fall ist sollte man nicht versuchen, ihn direkt zu löschen, da stattdessen der nächste davorstehende END-Befehl bzw. das nächste nicht-lokale Label gelöscht wird).

Mit dieser Technik lassen sich fast alle Bytes gewinnen:

0	Leerstelle nach löschen
1-159	durch entsprechende Befehle
128-227	aus indirekten Adressen 00-99
240-244	aus RCL IND T bis RCL IND L
241-255	aus entsprechenden Alphastrings

Mit etwas Geschick lassen sich diese Bytes dann zu neuen Befehlen zusammensetzen. Einige Schwierigkeiten treten dabei jedoch auf. Verändert man nämlich den Programmspeicher, werden sofort die Sprungpositionen in GTO- und XEQ-Befehlen gelöscht, Sprungbefehle mit fertigen Sprungpositionen können erst garnicht (ohne weiteres) entstehen. Dies wirft Schwierigkeiten auf, wenn etwa ein derartiger Befehl als Zwischenprodukt entsteht, z.B. beim Aufbau eines Alphastrings. END-Befehle als Zwischenprodukte sind ebenfalls unangenehm, da diese .. nach Ausführen von PACK im CAT 1 stehen und dann nicht mehr unbefangen weiterverarbeitet werden können. Solche Schwierigkeiten kann man wohl vermeiden, wenn man gewissermaßen "unter einem Alphastring versteckt" bastelt und erst zum Schluß den "Korken" entfernt. Dies ist aber sehr kompliziert, da man die zwischendurch erzeugten Befehle auch nicht mehr durch PACK zusammenschieben kann. In solchen Fällen ist es einfacher, auf das

"BASTEL"-Programm zurückgreifen (falls möglich).
Etwas spitzfindiger muß man auch bei der Erzeugung der Bytes 228 bis 239 sein, z.B. in dem Befehl RCL IND A:

1) LBL A, ENTER, ENTER eingeben, RCL b auf den 2. ENTER
2) ENTER-Befehle löschen, RCL IND Y, GTO A eingeben
3) GTO A mit SST im RUN-Modus ausführen
4) kurz auf PRGM schalten, danach im RUN-Modus STO b
5) auf PRGM schalten und folgende Befehle eingeben:
   RCL IND Y, ENTER, RCL IND Z, RCL IND 16 (IND 16 für RCL)

Unter anderem ist nun der Befehl RCL IND A entstanden.
Zu 4) Nach Durchführung von GTO, GTO. , PACK u.v.a.m. verliert der HP-41C die Orientierung im Programmspeicher und orientiert sich deshalb bei Umschalten auf PRGM neu, so daß der "Schwindel" bemerkt wird. Deshalb vor STO b kurz auf PRGM schalten.

Der Vorteil dieser Bastelmethode besteht darin, daß sie auch im Key-Assignment anwendbar ist, wenn man dessen Aufbau kennt. Man kann die zu belegenden Funktionen an der richtigen Stelle aufbauen, ggf. auch den Alphastring der Länge 0. Dieser ist sehr wichtig, da sonst dieses und alle davor stehenden Key-Register ignoriert werden. Dieser String muß unter Benutzung des davor stehenden Registers auch dann aufgebaut werden, wenn man das Register durch das Bastelprogramm erzeugt und dem entsprechenden STO IND-Befehl in das Key-Assignment gebracht hat, da ja stattdessen eine 0 am Anfang des Registers steht. Nützlich zum Basteln im Key-Assignment ist auch das JMP-Programm: mit dem aus 192,3 erzeugten Alphastring kann man etwa mittels STO b zum 4. Byte des zuletzt belegten Registers springen, man kann den String dann auch bis zur weiteren Verwendung abspeichern.

Wichtig: Will man im Key-Assignment Schritte eingeben, so muß man ggf. andere Schritte löschen und Leerstellen schaffen, da sonst Memory lost erscheint. Besondere Vorsicht ist bei Zahleneingaben geboten, da in diesem Falle auch vor und hinter der Zahl ein Leerbyte stehen muß. Außerdem ist zu beachten, daß durch PACK die einzelnen Befehle hier nicht zusammengeschoben werden, dafür aber auch nicht die Sprungpositionen in GTO- und XEQ-Befehlen gelöscht werden.

Man beachte auch, daß die angegebenen Tastenbelegungen mit den Tastenbelegungsflags übereinstimmen, was z.B. dadurch gewährleistet ist, daß man die entsprechenden Tasten vorher

belegt und in diesem Register bastelt. Dabei ist zu beachten: Wird eine Taste belegt, so wird die Belegung vorrangig in die noch freie Hälfte eines bisher nur halb genutzten Key-Registers geschrieben, ansonsten in das vom Programmspeicher gesehen letzte Register (für das JMP-Programm mit 192 zu adressieren). Übereinstimmung mit den Tastenbelegungsflags läßt sich im Zweifelsfalle auch dadurch erreichen, daß man eine Status-Karte beschriftet und wieder einliest. Ähnlich kann man auch in das vierte Byte eines Alphanumerischen Labels eine Tastenbelegung schreiben, das Programm auf Karte aufzeichnen und im USER wieder einlesen. Auf diese Weise ist es sogar möglich, die SHIFT-Taste (bzw. die geshiftete SHIFT-Taste!) und die imaginäre zweite Hälfte der ENTER-Taste zu belegen.

PRIVATE



Und last not least: der Sprung in ROM's! Anleitung:

1) Zwei Bytes in R01 und R02 eingeben
2) XEQ "CSTR"
3) In irgendein ROM an irgendeine Stelle gehen, kurz auf PRGM schalten
4) den durch "CSTR" erhaltenen String in b abspeichern und wieder auf PRGM schalten, fertig!

Zu 1) Die beiden Bytes haben folgende Bedeutung:

0/ 0	bis	47/255	12-K-ROM-Bereich des HP-41C
96/ 0	bis	111/255	Drucker
128/ 0	bis	143/255	ROM in Position 1

160/0 bis 175/255 ROM in Position 2
192/0 bis 207/255 ROM in Position 3
224/0 bis 239/255 ROM in Position 4

Kein Wunder also auch, daß der Drucker stets zuerst im CAT 2 erscheint! Wenn man jetzt die Befehle des Microprocessors kennt kann man also mit einigem Arbeitsaufwand die ROM-Programme entschlüsseln.

Das waren also in etwa meine Erfahrungen, die ich gesammelt und die Techniken, die ich entwickelt habe. Mit etwas Einfallsreichtum und ein wenig Glück lassen sich sicher noch tausende von Tricks entdecken. Ein Beispiel: Ist während des Programmablaufs der Flag 52 (PRGM-Modus) gesetzt und kommen in diesem Programm Zahlenbefehle vor, so fängt der Rechner plötzlich an, sich selbst Zahlenbefehle einzugeben! Einfaches Beispiel: XEQ "ABC", 5.
Noch ein Wort zur Umwandlung von Daten- in Programmregister. Diese läßt sich auch ohne Benutzung des STO IND-Befehls erreichen. Fügt man einen Befehl in den Programmspeicher ein, ohne daß die erforderlich Anzahl von Leerbytes vorhanden ist, wird ein ganzes Register eingefügt und der Rest des Programmspeichers nach oben geschoben. Gibt man aber etwa hinter dem letzten Byte des Datenregisters R01 einen Befehl ein (mit dem JMP-Programm kann man leicht hierher gelangen, man muß bloß die Nummer des Registers berechnen), so wird das Datenregister R00 zum ersten Programmregister. Dies kann man sicher für ein modifiziertes BASTEL-Programm ausnutzen.

Zu sagen bliebe noch, daß ich selbstverständlich in den Anwender-Club eintrete und um die bisherigen Veröffentlichungen bitte. Weiter bitte ich darum, mir die WALL-Karten oder die entsprechende Zahl leerer Karten zurückzuschicken.
Ansonsten viel Spaß beim Basteln!!!

Matthias Grabiak

Meine Adresse:

Matthias Grabiak
[illegible]-Str. [illegible]
6370 Oberursel/Ts.
Telefon 06171/[illegible]

Lieber Oliver! Oberursel, den 14. 9. 80

Mit Dank habe ich Deinen Brief erhalten. Zunächst ein Wort zu der entfleuchten Seite. Als ich sie ein paar Tage nach Verschicken des Briefes fand, war ich selbst erschrokken, da sie nämlich besonders wichtig war. Diesem Brief aber liegt sie nun bei.

In Deiner Antwort äußertest Du die Hoffnung, daß meine Aktivität auf dem Gebiet der synthetischen Programmierung (Basteln) weiter anhält. Z.T. schon in Erfüllung gegangen: Das CSTR-Programm wurde ersetzt durch ein BLD-Programm (in JMP u. CTLG nun XEQ"CSTR" ersetzen durch 2, XEQ"BLD"), das CNT-Programm wurde von den letzten beiden auf alle sieben Bytes des x-Registers erweitert, und außerdem habe ich ein BASTEL-Programm entworfen, das bei jedem HP-41C funktioniert, mit dem man nicht mehr auf 6 Bytes beschränkt ist und mit dem man überall und nicht nur an einer festgelegten Stelle basteln kann. Entscheidend war die Erkenntnis, daß Daten durch STO-Befehle allein noch nicht verändert werden (Dies ist auch in dem Bericht unter dem Punkt "Zur Datenspeicherung" zu korrigieren). Dieser Punkt war mir gar nicht aufgefallen, ich merkte es erst nach Studium des KA-Programms (Prisma, 18/7-80). Zu diesem Programm noch einige Bemerkungen:

1) Bei mir funktioniert es noch nicht so ganz

2) Es läßt sich teilweise vereinfachen. Die Schritte 12ff. lassen sich ersetzen durch einen Alphastring der Länge 5, nämlich durch STR(1,105,12,0,191), wenn ich diese Notation einführen darf, wobei ich gleichzeitig den Vorschlag mache, im Zweifelsfalle Informationen über Alphastrings unter Angabe der Schrittnummer in dieser Weise hinzuzufügen. Die Schritte 24-26 des KA-Programms sind durch STR(240) zu ersetzen, wodurch bereits das Unterprogramm B überflüssig wird. Schritt 155-160: STR(127,0). Ergänzend zu dem Bericht von Detlev Book: XROM 00,nn ist gleichwertig mit XROM nn,nn. XROM nn,00 ist der Name des entsprechenden Moduls!

Ein Wort zu Prisma 17/7-80: Springen zu einem Programmschritt durch Manipulation des e-Registers ist nicht möglich, entscheidende ist das b-Register.

Doch nun zur Beschreibung der neuen Programme.

BLD (auf Taste-25)

Mit diesem Programm kann jeder beliebige Alphastring bis zur Länge 7 erzeugt werden.

Eingabe: Bytes des zu erzeugenden Strings in R01 ff, Länge des Strings in das x-Register.

Das Programm läßt die Register 01-07 unverändert, das Stringzeichen richtet sich nach dem 256-ger-Rest des Absolutbetrages des jeweiligen Registers.

Ausgabe: String in Alpha-Register, erste 6 Zeichen in x und R00; y-Register: durch den String erzeugter Inhalt des M-Registers

Häufige Anwendung: als Unterprogramm

Es liegen zwei Versionen bei. Die erste kann jederzeit verwendet werden, die zweite (BLD, Drucker) ist einfacher und arbeitet schneller, greift aber auf den Drucker-Befehl BLDSPEC zurück.

CET (auf Taste -63)

Gegenteil von BLD. Jedes einzelne Byte des x-Registers wird analysiert, die Ergebnisse in R01 bis R07 abgespeichert und am Ende außerdem einzeln angezeigt. Nach jeder Anzeige auf R/S drücken, auch nach dem 7. Byte, damit der ursprüngliche Status erhalten bleibt. Bei gesetzten Flag 00 wird die Anzeige unterdrückt, was nützlich ist, wenn das CET-Programm als Unterprogramm aufgerufen werden soll.

BASTEL (auf Taste -74; greift auf BLD zurück)

Dieses Programm ermöglicht es, jede beliebige Folge von Bytes an irgendeine Stelle im Programmspeicher zu schreiben; Voraussetzung zur Angabe dieser Stelle ist, daß eine Taste mit RCL b belegt ist.

Kurzbeschreibung: Den Nippel durch die Lasche ziehen!

Ausführliche Anleitung: Man gehe zu der betreffenden Stelle im Programmspeicher und gebe mindestens 13 neutrale 1-Byte-Befehle ein (ENTER), falls die Anzahl der zu synthetisierenden Bytes 7 überschreitet jeweils 7 weitere Befehle für jedes angefangene Register. Dann gehe man zu dem ersten 1-Byte-Befehl zurück und führe RCL b aus, daraufhin XEQ"BASTEL". Das b-Register ist bekanntlich entscheidend für die Position im Programmspeicher. In dem BASTEL-Programm verarbeitet der Re-

chner diese Information zu einem Alphastring, der in R09 abgespeichert wird. Durch RCL 09, STO b kann man daher jederzeit zur "Baustelle" zurückkehren. Auch der Inhalt von R08 darf nicht verändert werden, da sich hier die Information befindet, welches Register als nächstes mit den angegebenen Bytes beschriftet wird. Das zuerst beschriftete Register ist das auf die angegebene Position folgende, wenn sich die Position aber am Anfang eines Registers befindet dieses Register selbst. Auf jeden Fall wird kein Befehl überschrieben, der sich vor der angegebenen Position befindet.

Nach XEQ"BASTEL" kommt der Rechner in eine Eingabeschleife. Man gibt hier ohne R/S die zu synthetisierenden Bytes ein. Der Rechner zeigt die letzten Eingaben an, bei angeschlossenem Drucker werden sie außerdem ausgedruckt. Nach jeweils 7 Bytes ein Register synthetisiert und in den Programmspeicher geschrieben. Besondere Eingaben:

negatives Vorzeichen	letztes zu synthetisierendes Byte (-256 bei 0)
XEQ A	Abbruch der Eingabe ohne Beschriftung des nächsten Registers
XEQ B	Zu synthetisierende Bytes bereits in R01-R07 eingegeben.

Nach Anhalten des Programms kann man mittels STO b oder SST zu dem ersten Bastelregister springen (nach SST ist die Nummer des Programmschritts nicht korrekt). XEQ"BC" erlaubt es, die Eingabe fortzusetzen, es werden dann die den bisher beschrifteten Registern folgenden übersohrieben. Falls man erneut von dem ersten Bastelregister aus starten möchte: RCL09, XEQ"BASTEL".

Anwendung des BASTEL-Programms als Key-Assignment-Programm:

1) Dafür sorgen, daß vorher eine gerade Zahl von Tasten mit Funktionen belegt sind (Irgendeine Taste belegen, Belegung wieder löschen, falls sich die Anzahl der freien Register verringert hat).
2) Gewünschte Tasten mit irgendwelchen Funktionen belegen.
3) 192+Anzahl der zu synthetisierenden Key-Register in R01 eingeben, 1, XEQ"BLD"
4) XEQ"BASTEL"
5) Wie oben beschrieben Bytes der Key-Register eingeben
6) Ggf. zusätzliche Belegung nach 1) löschen

Inzwischen habe ich auch eine neue manuelle Bastelmethode erfunden, nämlich mit Hilfe des "Wolf-Befehls". Diesen kann man auf eine Taste legen, wenn man für die beiden Bytes zur Bestimmung der Funktion 247 und 87 eingibt (statt 87 auch alles andere außer 0 möglich; 87 steht für W wie Wolf). Hier die Anleitung:

"Wolf-Befehl"

Dieser Befehl entspricht den beiden Bytes 247 und 87 als 2-Byte-Key-Assignment. Im Programm-Modus bewirkt er die Eingabe eines Alphastrings der Länge 7, er wird aber vom HP-41C als 3-Byte-Befehl gehalten. Ist beim Eingeben aber keine ausreichende Lücke vorhanden, können bis zu 5 Bytes von dem Alphastring "aufgefressen" werden. Gibt man den Wolf z.B. zwischen zwei ohne Leerbyte aufeinanderfolgende Befehle ein, so wird zunächst ein Registers eingefügt, an dessen Anfang der Wolf-Befehl geschrieben wird. Dieser umfaßt aber insgesamt 8 Bytes, so daß das erste Byte des folgenden Befehls von dem String absorbiert wird. Man kann dies wieder rückgängig machen, indem man den String seinerseits wieder auf die gleiche Weise durch den Wolf-Befehl zerstört. Die Handhabung des Wolf-Befehls als Bastel-Instrument ist ähnlich wie die von RCL b/STO b. Hier ein Beispiel:

1) RCL IND 17, RDN eingeben
2) zweimal BST
3) Wolf eingeben
4) Wolf löschen

Mit dem Wolf hat man gleichzeitig das RCL von RCL IND 17 gelöscht, zurück bleibt STO M.

Soweit der neuste Stand meines Bastel-Labors. Anwendungsbeispiele werden noch folgen, aber frühestens Mitte Oktober. Damit ich nicht als einseitiger Bastel-Fanatiker verschrien werde, habe ich noch drei Programme beigelegt, die (fast) nur aus normalen Befehlen bestehen.

1) PFZ: Zahl eingeben, XEQ"PFZ", Rechner druckt Primfaktorenzerlegung aus.
2) VSZE: Rechner bestimmt Anzahl der Datenregister und zeigt diese an. Die Geschwindigkeit ist etwa dieselbe, egal ob 0 oder 319 Datenregister vorhanden sind, da eine richtige Ratestrategie einprogrammiert ist. x- und y-Register werden durch das Programm nicht verändert.
3) ∫I: Integralprogramm

Nach XEQ"∫I" alle Abfragen beantworten und R/S drücken. Eingabe n bei Schritte/1 bedeutet n Schritte bei einem Integrationsintervall der Länge 2, 2n Schritte bei einem der Länge 2 usw. Das Ergebnis der Integration wird angezeigt und ist außerdem in R12 gespeichert. Nach Ausführung der Integration bestehen folgende Möglichkeiten:

XEQ a oder einfach R/S nach Eingabe einer Zahl	Selbes Intervall, aber neuer Wert für Schritte/1
Zahleneingabe, XEQ A oder nichtlokal XEQ"IF"	Stammfunktion (zuerst eingegebene Untergrenze, aber neue Obergrenze)
XEQ 01	Selbe Funktion, völlig neue Angaben

Vorteil des Programms: Macht man nach XEQ"∫I" alle Angaben außer der Obergrenze und gibt, falls ≠ 0, den Funktionswert an der eingegebenen Untergrenze in R12 ein, kann man durch XEQ "IF" oder lokal XEQ A die Stammfunktion eines Wertes berechnen, sie vom Drucker plotten lassen oder auch mit entsprechenden Programmen die Nullstellen suchen lassen.

Zu einer ausführlicheren Beschreibung der Programme habe ich momentan leider keine Zeit.

Noch ein anderes Thema: An sich braucht man bei der Programmierung des HP-41C mit den Bytes nicht besonders sparsam zu sein. Eine Schallmauer ist aber manchmal durch die Anzahl der beschrifteten Magnetkartenspuren gegeben. Deshalb hier ein paar Möglichkeiten zum Einsparen von Bytes:

1) Benutzung der Flags 42, 48 und notfalls 50. Diese lassen sich nämlich durch die 1-Byte-Befehle GRAD/DEG, AON/AOFF und AVIEW/ CLD steuern.

2) Man spart sich ein Byte, wenn man statt RCL L den Befehl LAST X benutzt; aber auch beim Speichern in das L-Register läßt sich ein Byte ein Byte einsparen, wenn man statt STO L den Befehl SIGN benutzt (auch bei Alphastrings möglich!)

3) Manchmal kann man zwei Abfragen hintereinanderschalten. Der folgende Befehl wird genau dann ausgeführt, wenn die erste Bedingung <u>nicht</u> erfüllt ist oder wenn die zweite erfüllt ist oder wenn beides der Fall ist. Z.B. x≠y? x>y? entspricht der nichtvorhandenen Abfrage x≥y?.

Das mag ■ den Augenblick genügen. Statt leeren Magnetkarten hätte ich diesmal gerne die beweglichen Feiertage sowie die U-Boot-Jagd. Ansonsten wünsche ich noch herzliches

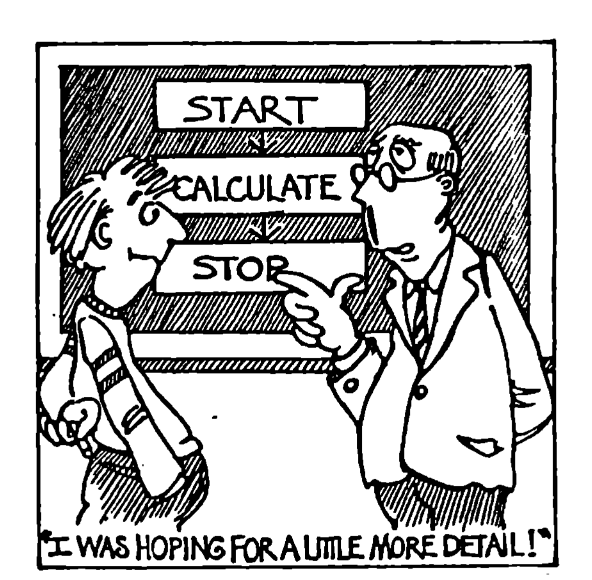
Happy
Programming

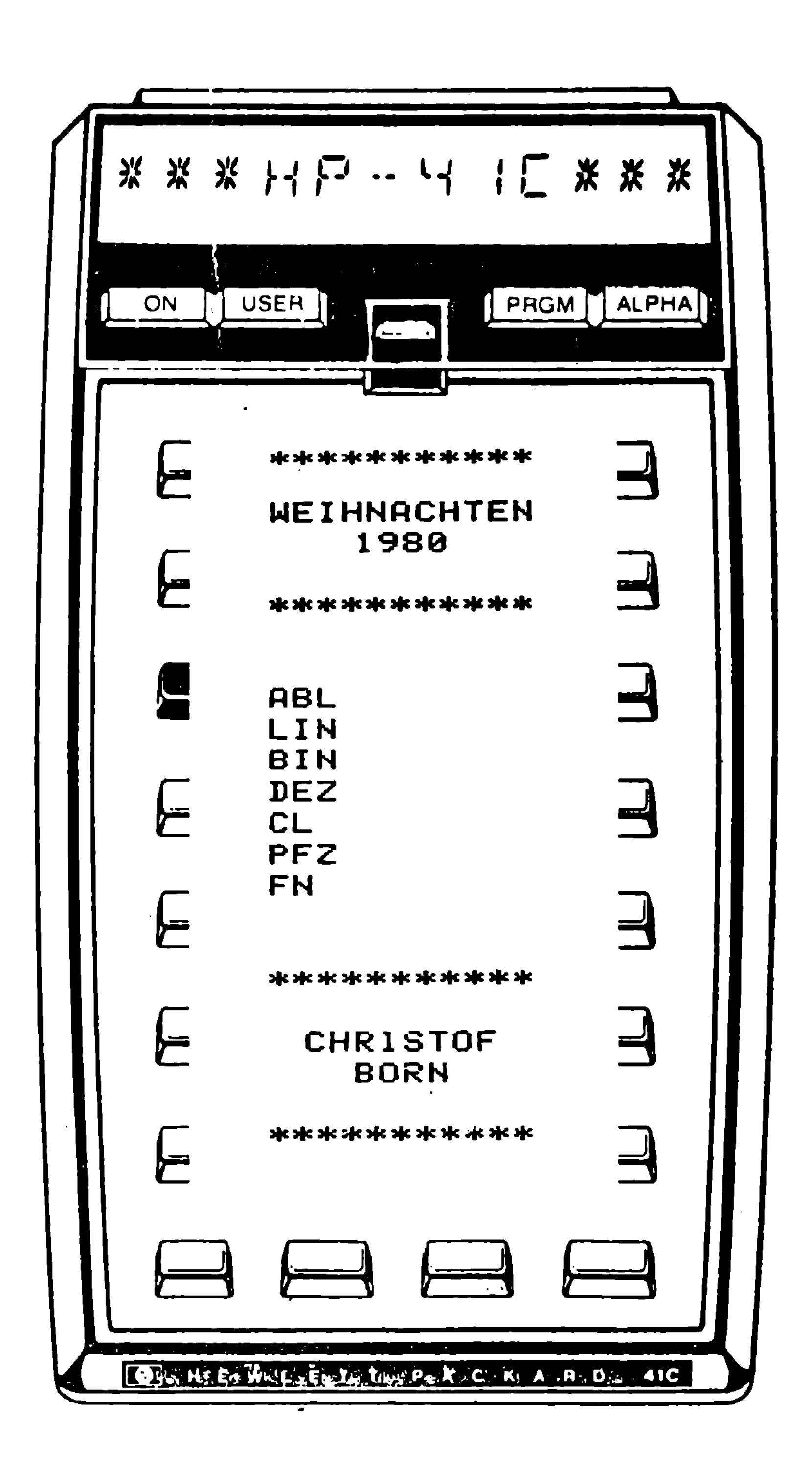
und noch mehr Spaß beim Basteln!!!

Matthias

Anmerkung:
Ich möchte Matthias zu diesen beiden gelungenen Beiträgen herzlich gratulieren und hoffe , daß viele Clubmitglieder viel Spaß daran haben werden .
Zumal diese beiden Berichte aber mehr oder weniger "Programmbeschreibungen" sind,ist es sicherlich notwendig,die entsprechenden Programme auf Magnetkarte zu besitzen.Ob die wichtigsten Teile auch als Bar-Codes im nächsten Quartal erscheinen,steht nicht fest.
Daher : wersich mehr für dieses Bastelprogramm und alles,was dazugehört,interessiert,schicke bitte zehn Magnetkarten an Walter Pieperhoff (2Ø5) und lege einen adressierten,freigemachten Rückumschlag und eine 5Ø Pfg Briefmarke bei .
Ebenfalls von Walter ist ein wenig überarbeitetes Key-Assignment-Programm erhältlich (zwei Magnetkarten) . Alle neuen Mitglieder oder diejenigen , die jetzt erst ihren Kartenleser bekommen haben , sollten dises Progeamm anfordern , denn es stellt die zentrale Rolle in der synthetischen Programmierung dar . Außerdem wird es im nächsten Quartal in überarbeiteter Form als Bar-Code erscheinen.

Fröhliches Basteln 　　　　Euer
Oliver





HP-41C
ON
USER
PRGM
ALPHA
****
WEIHNACHTEN
1980
****
ABL
LIN
BIN
DEZ
CL
PFZ
FN
****
CHRISTOF
BORN
****
41C

Tabelle der Programme:

Name:	Datenreg.:	Prgm.Reg.:	Zeilen:	benötigte Label's:
ABL	3	7	23	ABL
LIN	7	17	71	LIN, A, o1-o3
BIN	3	15	54	BIN, oo-o5
DEZ	5	13	44	DEZ, o1-o4
CL	o	3	11	CL
PFZ	3	26	113	PFZ, o1-o8
FN	5	3o	114	FN, A-J, o1-o6

ABL leitet bekannte Funktionen an beliebiger Stelle ab.

LIN berechnet Funktionsgleichung einer Geraden durch zwei gegebene Punkte und deren Abstand.

BIN wandelt Dezimalzahl in Binärzahl um.

DEZ wandelt Binärzahl in Dezimalzahl um.

CL bringt den Rechner in normalen Status zurück.

PFZ zerlegt eine beliebige Zahl in ihre Primfaktoren.

FN zeigt die x- und f(x)-Werte einer Funktion an; mit Fehlererkennung und 1o Sonderfunktionen.

Ich möchte darauf hinweisen, daß es möglich ist, daß die Programme "PFZ" und "FN" in der nächsten Zeit in der Zeitschrift "Chip" abgedruckt werden! Es ist aber nicht sicher.

Zum Programm ABL:

Dieses Programm leitet - wie bereits gesagt - eine Funktion an einer beliebigen Stelle ab, bestimmt also dort ihre Steigung, d.h. die Steigung des Graphen.
Eigentlich müßte dazu der Grenzwert des Differenzenquotienten gefunden werden, also $\lim_{h-o} \left(\frac{f(x+h)-f(x)}{h} \right)$. Dies wird in der Differentialrechnung aber gern als Näherung $\frac{dy}{dx}$ dargestellt. Dabei soll das h einen möglichst kleinen Wert haben, muß aber nicht gegen Null gehen. Im Programm habe ich für h $1o^{-5}$ gewählt, was in der Regel ausreichen dürfte. Das Programm hat allerdings die Voraussetzung, daß die Funktion an der Stelle x überhaupt differenzierbar ist!

Input/Output:

Nach XEQ 'ABL erscheint in der Anzeige "FN?"; darauf erwartet der Rechner den Namen des Labels, unter dem die entsprechende Funktion abgespeichert wurde. Nach R/S erscheint "X?" woraufhin die Stelle eingegeben werden muß, an der die Funktion abgeleitet werden soll. Nach nochmaligem R/S erscheint z.B. "M=o,5ooo" was die Steigung angibt.

Die Statusinformationen entnehmen Sie bitte aus der voranstehenden Tabelle auf Seite 2.

Zum Programm LIN:

Das Programm LIN berechnet die Funktionsgleichung einer Geraden durch zwei gegebene Punkte $P_1(x_1/f(x_1))$und $P_2(x_2/f(x_2))$.
Es wird auch deren Abstand ermittelt.
Die allgemeine Funktionsgleichung einer Geraden ist f(x)=mx+b.
Das m (die Steigung) wird berechnet als $m=(f(x_2)-f(x_1))/(x_2-x_1)$.
Das b (der Schnittpunkt mit der f(x)-Achse wird berechnet als $b=f(x_2)-m*x_2=f(x_1)-m*x_1$. Als Abstand der beiden Punkte ergibt sich nach Pythagoras: $d=SQRT((x_2-x_1)^2+(f(x_2)-f(x_1))^2)$.
Da in der Regel eine Ausgabe wie "Y=1,ooX+o,oo" unerwünscht ist,
wird wenn m=1 ist, m weggelassen,

wenn b=o ist, b weggelassen,

wenn b negativ ist "+" weggelassen,

so daß im obigen Fall das Resultat "Y=X" wäre.
Das Programm erkennt auch ob es sich bei der Funktion um eine Parallele zur x-Achse handelt, m also o ist. In diesem Fall wird hinter "Y=" nur der Abstand mit Votzeichen angezeigt.

Input/Output:

Nach XEQ'LIN erscheint in der Anzeige "X1=" für den ersten x-Wert, nach R/S "Y1=" für den ersten f(x)-Wert, nach R/S "X2=" für den zweiten x-Wert und nach nochmaligem R/S "Y2=" für den zweiten f(x)-Wert. Die entsprechenden Werte sind jeweils einzugeben.
Nach Drücken von R/S kommt nun die Funktionsgleichung in die Anzeige. Ohne Überlauf der Anzeige können m und b maximal 99,99 werden. Nach R/S wird die Entfernung der beiden Punkte durch "DIST=..." angegeben.
Durch doppeltes R/S kann das Programm neu gestartet werden.

Zum Programm BIN:

Das Programm wandelt eine Dezimalzahl in eine Binär- bzw. Dualzahl um. Die Zahl, die umgewandelt werden soll, muß kleiner als 1o24 sein. Die Ausgabe der Binärzahl ist so geordnet, daß die Zahl von rechts nach links gelesen werden muß. Es stehen also die Stellenwerte in folgender Reihenfolge: ...,16,8,4,2,1.
Das Programm untersucht zuerst, ob die Zahl zur Verarbeitung klein genug ist. Ist sie das nicht, erfolgt die Ausgabe "ZU GROSS". Ist sie es aber, so versucht der Rechner zuerst 2^9 zu subtrahieren. Gelingt das, so führt er diesen Schritt aus und addiert $1o^9$ zu dem (im Moment noch nicht vorhandenen) "Binärwert", dekrementiert den Exponenten (9) und wiederholt den Vorgang - nur mit dem Unterschied, daß der Exponent bei 2^x und $1o^x$ nun um eins kleiner ist.
Sollte die Subtraktion nicht gelingen, so wird der Exponent erniedrigt, ohne daß der "Binärwert" verändert wird.
Da die nächstfolgende Addition in den "Binärwert"-speicher um eine Zehnerstelle kleiner ist, entsteht eine Null in der Reihe der Ziffern in ihm.
"11o1" heißt also, daß die ersten beiden und die letzte Subtraktion gelungen sind, die vorletzte aber nicht.
Die so entstandene, maximal zehnstellige Zahl wird vom Rechner dezimal aufgefaßt, gestattet aber dank ihrer Darstellung nur aus Einsen und Nullen sie als Binärwert zu lesen.

Input/Output:

Nach KEQ'BIN fragt der Rechner mit "DEZ=" nach dem umzuwandeln - den Dezimalwert. Nach Eingabe und R/S folgt die Ausgabe der von rechts nach links zu lesenden Binärzahl.

Zum Programm DEZ:

Das Programm wandelt eine Binärzahl in eine Dezimalzahl um. Zuerst prüft der Rechner, ob die Zahl zur Verarbeitung klein genug, d.h. kleiner als 1111111112 ist. Wenn nicht, erscheint die Anzeige "ZU GROSS".
Der Rechner "nimmt" nun die am weitesten rechts stehende Ziffer der Zahl, die von rechts nach links gelesen werden muß (!), die ja dem Binären Stellenwert 1 entspricht, multipliziert sie mit 2^0 - also 1 - und addiert den entsprechenden Wert zu Variablen d, die die umgewandelte Zahl darstellen soll. Der Exponent (im Moment noch o) wird um 1 erhöht, und es folgt die zweite Ziffer, die mit 2^1 multipliziert und dazuaddiert wird. Es wird also immer die n-te Ziffer mit 2^{n-1} multilpiziert bis n 1o erreicht. Dann muß - laut Vorbedingung darf die Zahl höchstens zehnstellig sein - die Zahl " zu Ende " sein.

Input/Output:

Nach XEQ'DEZ verlangt der Rechner mit "BIN=?" die Eingabe des umzuwandelnden Wertes. Nach R/S zeigt der Rechner die entsprechende Dezimalzahl in gewohnter Reihenfolge (die Einerstelle ganz rechts) an.

Zum Programm CL:

Das Programm CL ist zwar nicht sehr kompliziert, aber trotzdem recht praktisch.
Es ist dazu da, den Rechner wieder in den normalen Status zurückzubringen, nachdem er z.B. durch ein Programm in weniger zweckmäßige Modi gebracht wurde.
Das Programm löscht Stack,Register,alpha-Register,die Flags 24 und 25, setzt Flag 29, wählt das Anzeigeformat Fix 4 und das Winkelmaß DEG.
Es ist günstig, dieses Programm einer bestimmten Taste zuzuordnen, denn so können alle aufgezählten Funktionen mit einem Tastendruck ausgeführt werden.
Das Programm ist wie gesagt nicht sehr kompliziert, hat aber trotzdem zwei unübersehbare Vorteile:

1) das Programm benötigt keinen Kartenleser. Es ist klar, daß eine entsprechend beschriebene Statuskarte wesentlich besser die Aufgabe erfüllt, zum Normalstatus zurückzukehren, als dies Programm (da z.B. Size nicht programmierbar ist), aber es ist ebenso klar, daß die Möglichkeiten einer Statuskarte für HP-41c-Besitzer, die keinen Leser haben unerreichbar sind.
2) selbst wenn ein Kartenleser vorhanden ist, geht die Ausführung dieses Programms immer noch schneller als das Einlesen der Karte und ist vor allen Dingen schonender für den Kartenleser, der ja bekanntermaßen nicht allzu robust ist. Außerdem spart man mit diesem Programm den Batteriestrom für den Leser.

Zum Programm PFZ:

Das Programm zerlegt eine beliebige ganze Zahl größer als eins in ihre Primfaktoren. Sollte die Zahl diese Bedingungen nicht erfüllen, wird dies vom Rechner mit "NICHT ZERLEGBAR" angegeben. Der Rechner erkennt auch, ob es sich um eine Primzahl handelt. Zunächst wird die Zahl, falls sie kleiner als acht ist, gesondert untersucht. Das liegt daran, daß Wurzel aus sieben kleiner als der erste feste Testdivisor (drei) ist.
Dann wird die Zahl auf ihre Teilbarkeit durch zwei, dann drei, fünf, sieben, neun und so weiter untersucht. Ist sie teilbar, wird der entsprechende Divisor angezeigt, die Zahl durch ihn geteilt, und dieser Schritt so lange wiederholt, bis sie nicht mehr durch diesen Divisor teilbar ist. Dann wird der Divisor um zwei erhöht und wieder die Teilbarkeit geprüft. Das geht so lange, bis der Divisor größer als die Wurzel aus dem Anfangswert wird. Ist bis dahin noch keine Ausgabe eines Divisors bzw. noch keine Teilung erfolgt, muß es sich um eine Primzahl handeln. Das wird vom Rechner mit "PRIMZAHL" angezeigtgab es bereits eine Zerlegung, läuft das Programm normal weiter bis der Divisor größer als die Hälfte des Anfangswertes wird. Ein Weiterführen der Untersuchung über diesen Punkt hinaus wäre sinnlos, da dann der nächst Divisor kleiner als zwei sein müßte, was aber unmöglich ist.

Input/Output:

Drei Beispiele für den Dialog:

1.) für eine zerlegbare Zahl:
    XEQ'PFZ; in der Anzeige erscheint "Z=?".
    Nach Eingabe der Zahl (z.B. 9o) R/S.
    Nach ca. 5 sec ist die Ausgabe beendet (2,3,3,5,).
    Für 1o24 werden 1o, für 16oo 9sec benötigt.
2.) Für eine Primzahl:
    XEQ'PFZ; in der Anzeige erscheint "Z=?".
    Nach Eingabe einer Primzahl (zB. 157) erscheint in der Anzeige (hier nach ca. 5 sec) "PRIMZAHL"
3.) Für eine nicht zerlegbare Zahl:
    Der Rechner gibt nach knapp einer Sekunde "NICHT ZERLEGBAR"

Zum Programm FN:

Das Programm gibt nacheinander schrittweise die x- und f(x)-Werte einer zuvor eingegeben Funktion aus. Dabei werden Definitionslücken erkannt (wie z.B. x=o bei der Funktion 1/x).
Es ist zwar kein allzu kompliziertes und anspruchsvolles Programm, aber ich kann als Schüler der gymnasialen Oberstufe bestätigen, daß es zumindest in diesem Bersich sehr nützlich ist und oft angewandt wird.
Die doch etwas merkwürdige Struktur des Programms mag auf den ersten Blick unsinnig erscheinen, aber es ist auf dem HP-41c nicht anders möglich, die Anzeige auf gleiche Art mit einem einfacheren Programm darzustellen (es erscheinen dann immer am Ende der Ausgabe einige Zeichen, obwohl das wegen vorherigem FIX 2 und CLA's eigentlich nicht passieren dürfte).
Zur Fehlererkennung benutzt das Programm das Systemflag 25 das - zuvor gesetzt - nach einem mathematischen Fehler gelösbcht wird.

Input/Output:

Nachdem das Programm eingegeben ist, muß nach GTO.. noch die Funktion als Programm eingegeben werden, deren Werte berechnet werden sollen.
Hier zum Beispiel: GTO..

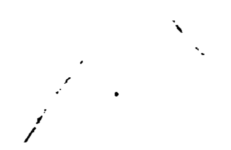
```
o1 LBL'1/X
o2 1/X
o3 RTN
o4 END
```

Nun kann das Programm mit XEQ'FN gestartet werden.
In der Anzeige erscheint "FN?". Der Rechner verlangt dadurch den Namen der zuvor eingegebenen Funktion. Nach der Eingabe: xxx R/S. In die Anzeige kommt "STEP?". Damit fragt der Rechner nach der Schrittweite, in der die einzelnen x-Werte folgen sollen. Nach Eingabe der Schrittweite (hier ,1) R/S.
Der Rechner fragt "AB?". Nun soll der Anfangswert für das erste x eingegeben werden. Hier z.B. -1. mit der nun folgenden Ausgabe "RUN/STEP?" fragt der Rechner, ob die Werte fortlaufend ohne anzuhalten (RUN) oder schrittweise (STEP) durchlaufen sollen.

Die Eingabe "2" für "STEP" ist dann sinnvoll, wenn man die einzelnen Werte zB. in eine Tabelle eintragen will.
Die Eingabe "1" für "RUN" ist dann zu empfehlen, wenn man sich nur ein Bild vom ungefähren Verlauf der Kurve machen will.
Bei "STEP" erwartet der Rechner nach jeder Ausgabe R/S.
Nach der Eingabe R/S drücken. Der Rechner beginnt nun mit der Programmausführung. Bei "RUN" folgen die Werte je nach Länge der zu berechnenden Funktion im Abstand von etwa einer Sekunde.
In der Anzeige erscheint (hier bei der Funktion 1/x):
"X1,00 F1,00". Dabei steht "X" für "X-Wert:" und "F" für "F(X)-Wert". Das Anzeigeformat ist auf zwei Nachkommastellen beschränkt, was aber in der Regel genügen dürfte.
Bei der Funktion 1/x erscheint bei x=o in der Anzeige:
"X0,00 ERROR". Das bedeutet, daß die Funktion für den betreffenden x-Wert nicht definiert ist.
Normalerweise würde das zur Fehlermeldung "DATA ERROR" führen nach der das Programm anhalten würde. Das wird aber hier umgangen, und der Rechner fährt mit dem nächsten x-Wert fort.
Das Programm läuft so lange weiter, bis es durch R/S gestoppt wird.
Soweit zum normalen Programmablauf. Die letzten 53 Zeilen (ab LBL A) wurden bislang noch nicht berücksichtigt.
Sie enthalten 1o Sonderfunktionen zur Programmausführung.
Diese sind den oberen zwei Tastenreihen zugeordnet; es empfiehlt sich, die einzelnen Funktionen auf der Magnetkarte oder einem Overlay zu markieren, da es sonst leicht zu unbeabsichtigten Funktionen kommt.
Sind diese Funktionen (oder einige davon) unerwünscht, so genügt es, die betreffenden Labels zu löschen. Der Programmrumpf bedarf dabei keiner Änderung.
Die Funktionen sind folgendermaßen auszuführen:
Wenn das Programm im "STEP"mode bei PROMPT anhält, bewirkt zB. der Druck auf die Taste "LOG": XEQ D. Die Funktion wird ausgeführt, und der Rechner kehrt selbsttätig zum Hauptprogramm zurück.
Im "RUN"mode ist es etwas schwieriger: kurz nach Erscheinen der Anzeige ist zuerst auf R/S zu drücken und dann auf die entsprechende Taste.

Die zehn Funktionen, die das Betrachten einer Kurve wesentlich angenehmer machn sind die folgenden:

Label:	Symbol:	Funktion:
A	↶	letzten Wert zurückrufen
B	Step*2	Schrittweite verdoppeln
C	Step/2	Schrittweite halbieren
D	→Run	Stepmode umwandeln in Runmode
E	→Step	Runmode umwandeln in Stepmode
F	X=	neue Eingabe für x-Wert
G	↔	Laufrichtung umkehren
H	-X	x-Wert mit (-1) multilpizieren
I	Step=X	Schrittweite gleich x-Wert setzen
J	Step=	neue Eingabe für Schrittweite



```
LBL'ABL
END          48 BYTES
LBL'CL
END          23 BYTES
LBL'BIN
END          99 BYTES
LBL'DEZ
END          92 BYTES
LBL'PFZ
END          179 BYTES
LBL'FN
END          213 BYTES
LBL'LIN
END          121 BYTES
```

```
01♦LBL "ABL"
"FN?" AON PROMPT
ASTO 00 AOFF "X?"
PROMPT STO 01 1 E-5 +
XEQ IND 00 ENTER↑
RCL 01 XEQ IND 00 -
1 E-5 / STO 02 "M="
ARCL 02 PROMPT END
```

```
01♦LBL "CL"
CLST STO L CLRG CLA
CF 24 CF 25 SF 29
FIX 4 DEG END
```

```
01♦LBL "BIN"
CF 29 FIX 0 CLRG
9,00001 STO 01 "DEZ="
PROMPT STO 00 1024
X<=Y? GTO 04

13♦LBL 00
2 ENTER↑ RCL 01 INT
Y↑X RCL 00 X<>Y X<=Y?
GTO 01 DSE 01 GTO 00
GTO 02

26♦LBL 01
RCL 01 INT 10↑X
ST+ 02 2 RCL 01 INT
Y↑X ST- 00 RCL 00
X=0? GTO 03 DSE 01
GTO 00

41♦LBL 02
RCL 00 X=0? GTO 03 1
ST+ 02

47♦LBL 03
VIEW 02 GTO 05

50♦LBL 04
"ZU GROSS" PROMPT

53♦LBL 05
END
```

```
01♦LBL "DEZ"
CF 29 CLRG ,01001
STO 01 "BIN=?" PROMPT
STO 02 1111111112
X<Y? GTO 03
12♦LBL 01
RCL 02 10 / FRC 10
* STO 04 ST- 02 10
ST/ 02 RCL 02 RCL 04
2 ENTER↑ RCL 01 INT
Y↑X * ST+ 03 RCL 02
X=0? GTO 02 ISG 01
GTO 01
37♦LBL 02
VIEW 03 GTO 04

40♦LBL 03
"ZU GROSS" PROMPT

43♦LBL 04
END
```

```
01♦LBL "PFZ"
FIX 0 CF 00 CLST
"Z=?" PROMPT STO 00
STO 01 INT RCL 01
X≠Y? GTO 05 RCL 00 2
X>Y? GTO 05 RCL 00 8
X>Y? GTO 08 CLA
22♦LBL 01
RCL 01 2 / ENTER↑
RCL 01 2 / INT X=Y?
GTO 03 3 STO 02
35♦LBL 02
RCL 01 RCL 02 /
ENTER↑ RCL 01 RCL 02
/ INT X=Y? GTO 04 2
ST+ 02 RCL 00 2 /
RCL 02 X>Y? GTO 06
FS? 00 GTO 02 RCL 02
RCL 00 SQRT X>Y?
GTO 02 CLA
62♦LBL 07
"PRIMZAHL" AVIEW
GTO 06
66♦LBL 05
"NICHT ZERLEGBAR" AVIEW
GTO 06
70♦LBL 03
"├2," AVIEW SF 00 2
ST/ 01 RCL 01 1 X=Y?
GTO 06 GTO 01
81♦LBL 04
"├ " ARCL 02 AVIEW
SF 00 RCL 02 ST/ 01
RCL 01 1 X=Y? GTO 06
GTO 02
93♦LBL 08
CLA RCL 00 2 - X=0?
GTO 07 RCL 00 2 /
ENTER↑ RCL 00 2 /
INT X=Y? GTO 01
GTO 07

111♦LBL 06
CF 00 END
```

```
01♦LBL "FN"
SF 27 FIX 2 "FN?" AON
PROMPT ASTO 00 AOFF
"STEP?" PROMPT STO 01
"AB?" PROMPT STO 02
"RUN/STEP?" PROMPT 3
+ STO 03
20♦LBL 01
SF 25 RCL 02
XEQ IND 00 STO 04
FS? 25 GTO 02 RCL 03
5 - X=0? GTO 06 CLA
"X " ARCL 02 "├ ERROR"
AVIEW GTO 03
38♦LBL 02
CLA "X" ARCL 02 "├ F"
ARCL 04 GTO IND 03
45♦LBL 03
RCL 01 ST+ 02 GTO 01
49♦LBL 04
AVIEW GTO 03
52♦LBL 05
PROMPT GTO 03
55♦LBL 06
CLA "X" ARCL 02
"├ ERROR" PROMPT
GTO 03
62♦LBL A
RCL 01 2 * ST- 02
GTO 03
68♦LBL B
RCL 01 2 ST* 01
GTO 03
73♦LBL C
RCL 01 2 ST/ 01
GTO 03
78♦LBL D
4 STO 03 GTO 03
82♦LBL E
5 STO 03 GTO 03
86♦LBL F
"X=?" PROMPT RCL 01 -
STO 02 GTO 03
93♦LBL G
RCL 01 CHS STO 01
GTO 03
98♦LBL H
RCL 02 CHS RCL 01 -
STO 02 GTO 03
105♦LBL I
RCL 02 STO 01 GTO 03
109♦LBL J
"STEP=?" PROMPT STO 01
GTO 03 END
```

```
01♦LBL "LIN"
FIX 2 "X1=?" PROMPT
STO 00 "Y1=?" PROMPT
STO 01 "X2=?" PROMPT
STO 02 "Y2=?" PROMPT
STO 03 RCL 01 -
RCL 02 RCL 00 - /
STO 04 RCL 00 *
RCL 01 - CHS STO 05
RCL 02 RCL 00 - X↑2
RCL 03 RCL 01 - X↑2
+ SQRT STO 06 "Y="
RCL 04 1 X=Y? GTO 01
RCL 04 X=0? GTO A
ARCL 04

48♦LBL 01
"├X"

50♦LBL A
RCL 05 SIGN -1 X=Y?
GTO 02 RCL 05 X=0?
GTO 03 "├+" ARCL 05
GTO 03

62♦LBL 02
ARCL 05

64♦LBL 03
FIX 2 PROMPT FIX 5
"DIST=" ARCL 06 PROMPT
END
```

Kommentar zu den Flußdiagrammen:

"BIN":

x: eingegebene Dezimalzahl
a: Laufwert zur Potenzierung
b: zu berechnender Binärwert

"PFZ":

z: eingegebene Zahl zur Zerlegung
a: Laufwert (Testdivisor)

"FN":

x: x-Wert
s: Schrittweite
m: Entscheidung für "RUN" (m=4) oder "STEP" (m=5)
Der Befehl "al:f(x)" bedeutet: die normale Ausgabe (ohne Fehleranzeige) wird in das alpha-Register geschrieben, aber noch noch nicht angezeigt. Das geschieht - je nachdem, ob m=4 oder m=5 ist - in Label o4 oder o5. Das Ausgabesymbol "ERROR" bedeutet: der Text mit der Fehleranzeige ".... ERROR" wird in das alpha-Register geschrieben und je nach den neben den Symbolen stehenden Befehlen per PROMPT oder per AVIEW ausgegeben.

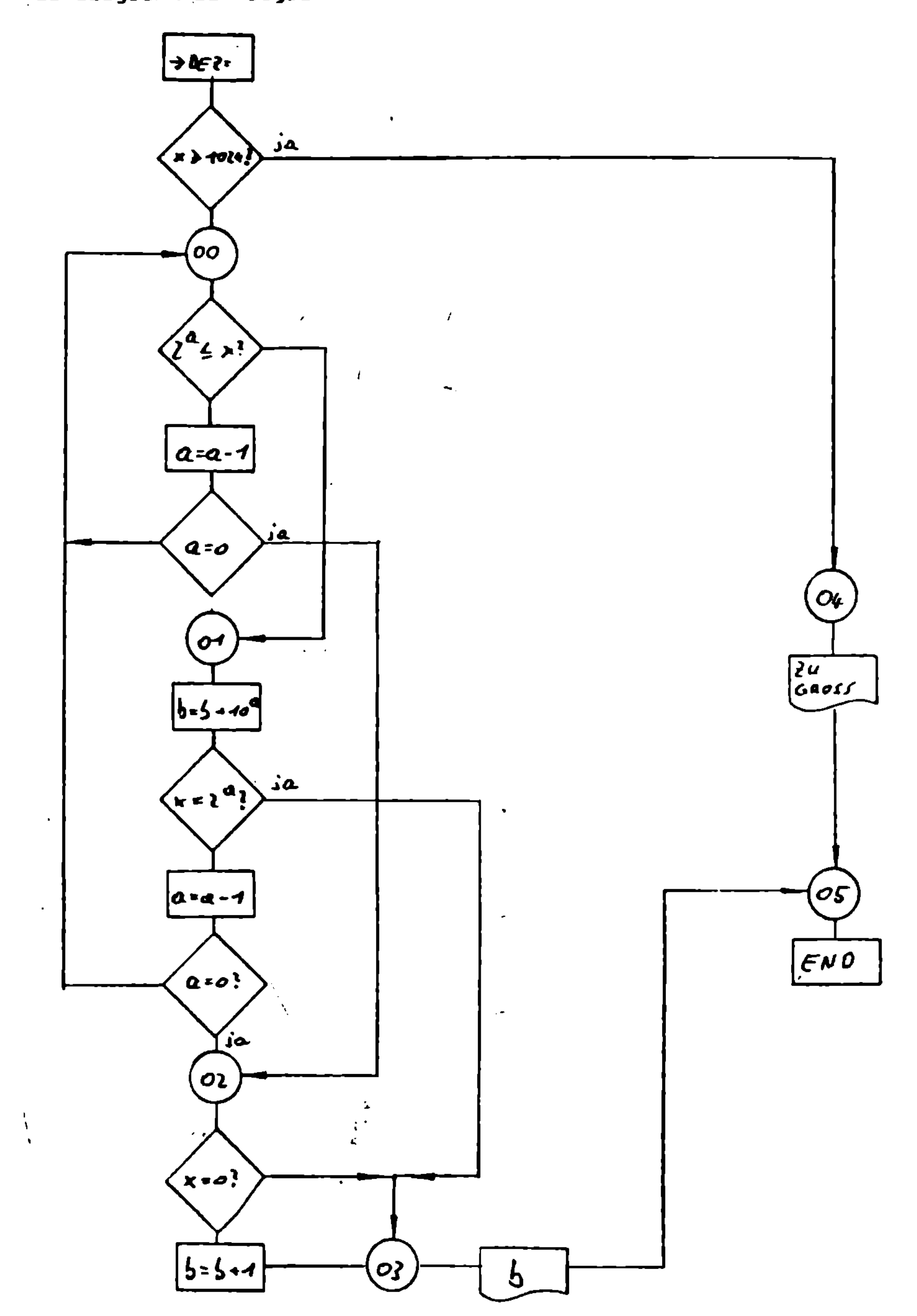
"DEZ":

x: eingegebene Zahl (binär)
z: am weitesten rechts stehende Ziffer von x
d: zu berechnender Dezimalwert

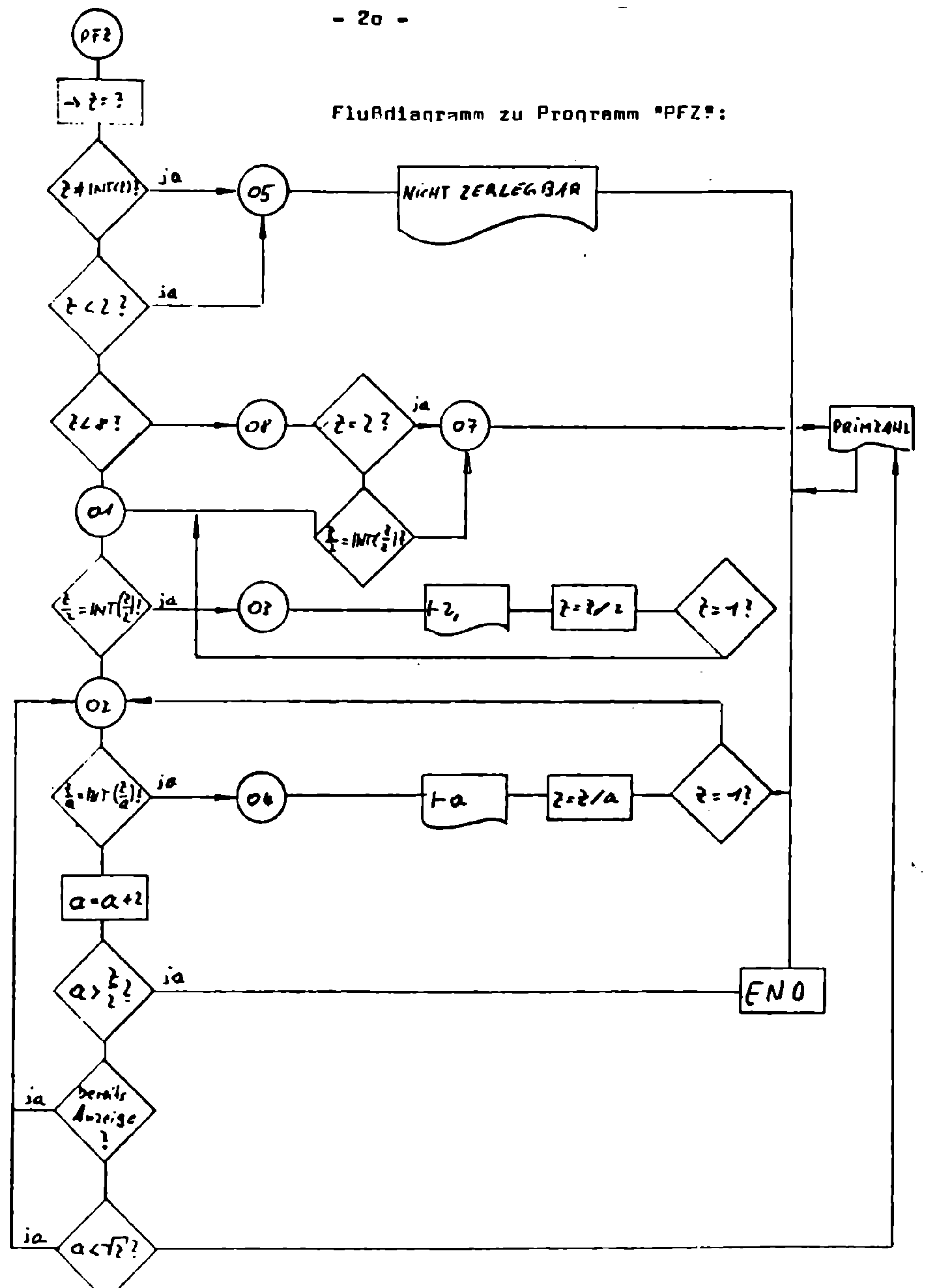
"LIN":

x1,y1,x2,y2: Koordinaten der beiden Punkte
m: Steigung der Geraden
b: Schnittpunkt der Geraden mit der y-Achse
d: Entfernung der beiden Punkte
Der Befehl "al:...." bedeutet: den entsprechenden Text ins alpha-Register schreiben, aber noch nicht ausgeben

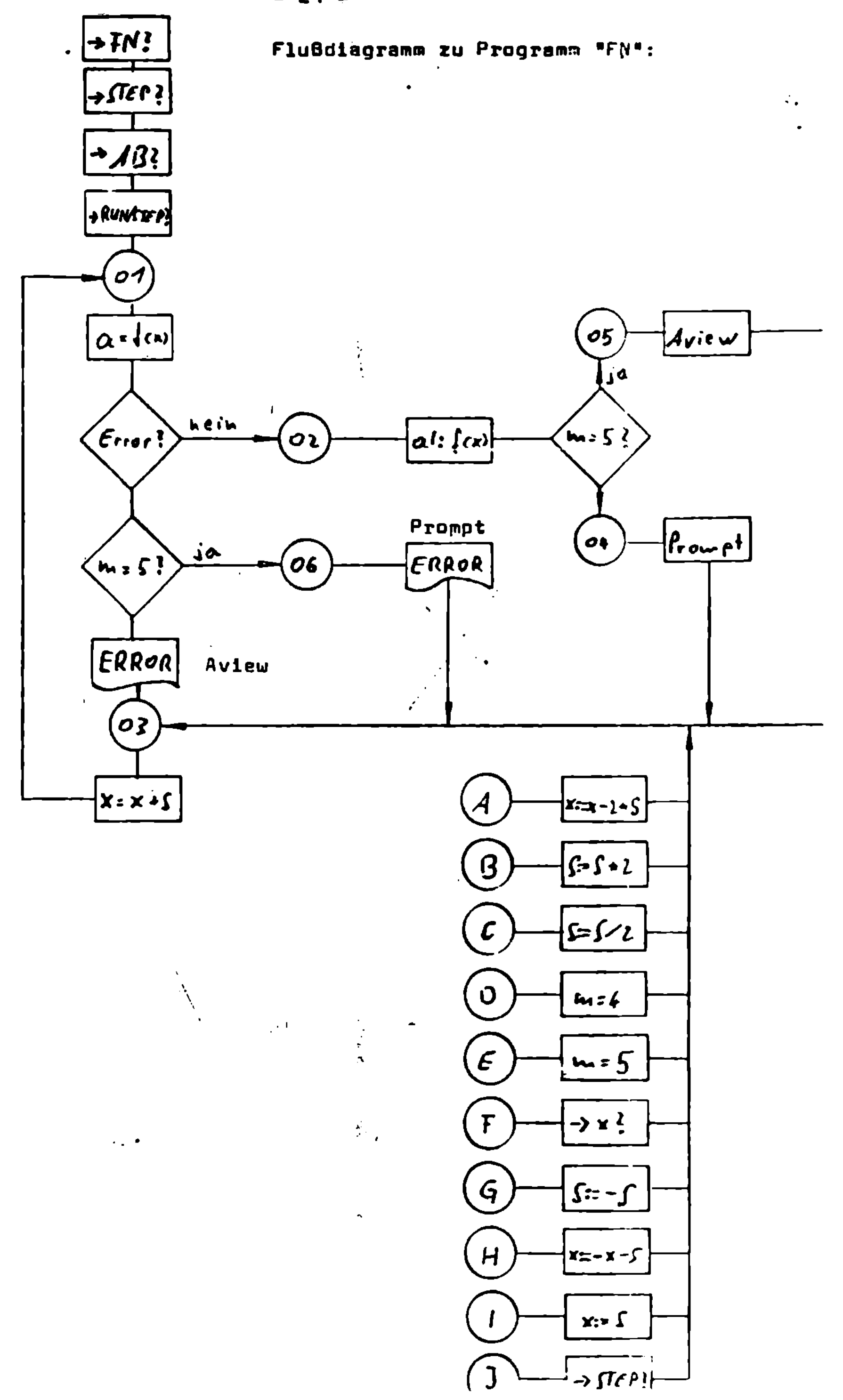
Flußdiagramm zu Programm "BIN":



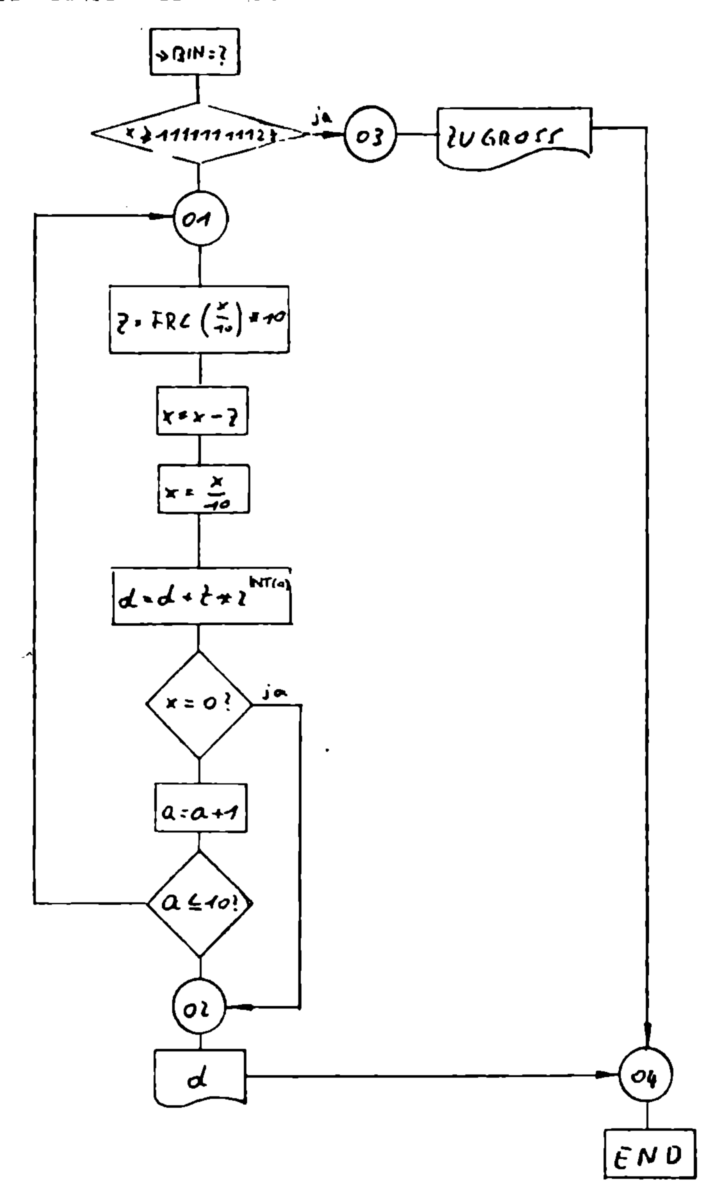
Flußdiagramm zu Programm "PFZ":



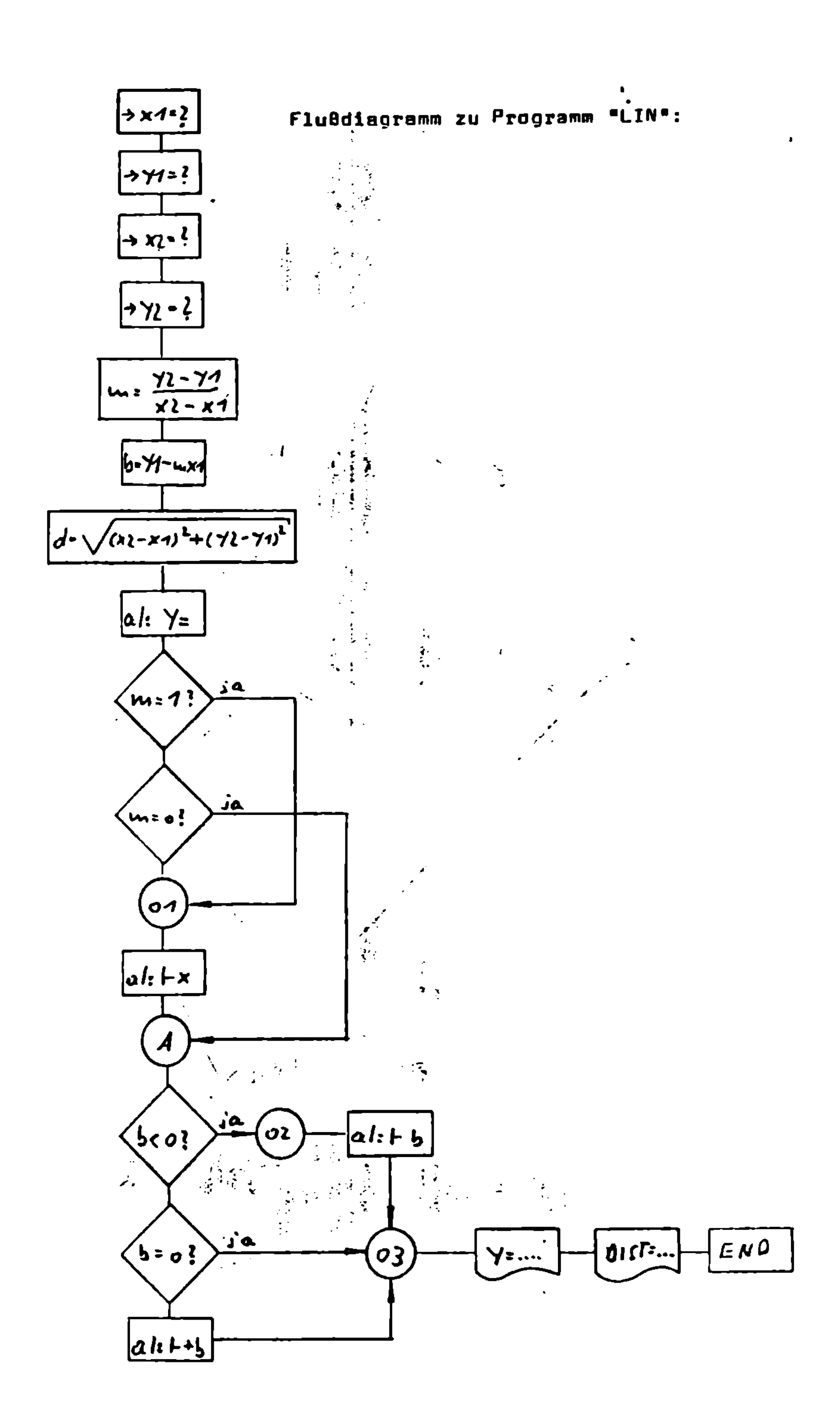
Flußdiagramm zu Programm "FN":



Flußdiagramm zu Programm "DEZ":



Flußdiagramm zu Programm "LIN":



Unterprogramm RUKUN: Differentialgleichungssystem nach Runge-Kutta

Gegeben seien N (gekoppelte) Differentialgleichungen erster Ordnung:

$$\frac{dy_i}{dx} = y_i' = f_i(x,\ y_1,\ \dots\ y_N) \qquad i = 1 \dots N$$

mit den N Anfangsbedingungen

$$y_i(x^{(\emptyset)}) = y_i^{(\emptyset)} .$$

Die Lösungen $y_i(x)$ werden -ausgehend von $x^{(\emptyset)}$ und $y_i^{(\emptyset)}$- punktweise durch Anwendung des Verfahrens von Runge-Kutta berechnet: zu einem kleinen Zuwachs h in x wird in vier Stufen ein kleiner Zuwachs q_i in y_i ermittelt. Der Index i läuft jeweils von 1 bis N.

1. Stufe: $x^{I} = x^{(\emptyset)}; \quad y_i^{I} = y_i^{(\emptyset)}$

$$k_i^{I} = \frac{h}{2} * f_i(x^{I},\ y_1^{I},\ \dots\ y_N^{I})$$

2. Stufe: $x^{II} = x^{(\emptyset)} + \frac{h}{2}; \quad y_i^{II} = y_i^{(\emptyset)} + k_i^{I}$

$$k_i^{II} = \frac{h}{2} * f_i(x^{II},\ y_1^{II},\ \dots\ y_N^{II})$$

3. Stufe: $x^{III} = x^{II}; \quad y_i^{III} = y_i^{(\emptyset)} + k_i^{II}$

$$k_i^{III} = \frac{h}{2} * f_i(x^{III},\ y_1^{III},\ \dots\ y_N^{III})$$

4. Stufe: $x^{IV} = x^{(\emptyset)} + h; \quad y_i^{IV} = y_i^{(\emptyset)} + 2 * k_i^{III}$

$$k_i^{IV} = \frac{h}{2} * f_i(x^{IV},\ y_1^{IV},\ \dots\ y_N^{IV})$$

Die neuen Punkte der Lösungsfunktionen ergeben sich mit

$$q_i = \frac{1}{3} * (k_i^{I} + 2 * k_i^{II} + 2 * k_i^{III} + k_i^{IV}) \quad \text{zu}$$

$$x^{(1)} = x^{(\emptyset)} + h; \quad y_i^{(1)} = y_i^{(\emptyset)} + q_i$$

Für den nächsten Schritt sind die $x^{(1)}$, $y_i^{(1)}$ wieder als Anfangswerte $x^{(\emptyset)}$, $y_i^{(\emptyset)}$ zu interpretieren, woraus sich dann wieder die nächsten Punkte $x^{(1)}$, $y_i^{(1)}$ ergeben usf.

RUKUN muß als Unterprogramm von einem Hauptprogramm aufgerufen werden, das folgende Aufgaben erfüllen soll:

1. Ggf. Zuweisung von $x^{(\emptyset)}$, $y_i^{(\emptyset)}$ und h zu den entsprechenden Registern (siehe Speicherplan)
2. Ggf. Zuweisung des Funktionsnamen für die f_i (s. u.)

)

an das Register $R_{Ø7}$.

3. Ggf. Erzeugung einer Schleife, die nur nach jedem m-ten Integrationsschritt Ergebnisse ausdrucken läßt.
4. Bildung der auszudruckenden Funktionswerte aus den $x^{(j)}$, $y_i^{(j)}$.
5. Anzeige bezw. Ausdruck der gewünschten Werte.
6. Beendigung des Programmes bei Erreichen eines Endwertes x_E.

Das Funktionensystem der f_i ($i = 1 \ldots N$) muß vom Benutzer programmiert werden, beginnend mit LBL "FN", endend mit END, wobei "FN" ein beliebiger Funktionsname (max. 6 Alpha-Zeichen) ist.

Speicherplan:

$R_{ØØ}$: Adresse für dynamische Speicherzuweisung

$R_{Ø1}$: Schleifenzähler für N

$R_{Ø2}$: S-1

$R_{Ø3}$: x^{I}; x^{II}; x^{III}; x^{IV}

$R_{Ø4}$: N

$R_{Ø5}$: Startadresse $S \geq 9$

$R_{Ø6}$: Inkrement h

$R_{Ø7}$: "FN"

$R_{Ø8}$: $x^{(Ø)}$; $x^{(1)}$

$R_S \ldots R_{S-1+N}$: $y_1^{(Ø)} \ldots y_N^{(Ø)}$; $y_1^{(1)} \ldots y_N^{(1)}$

$R_{S+N} \ldots R_{S-1+2N}$: $y_1' \ldots y_N'$

$R_{S+2N} \ldots R_{S-1+3N}$: $y_1^{I} \ldots y_N^{I}$; ... ; $y_1^{IV} \ldots y_N^{IV}$

$R_{S+3N} \ldots R_{S-1+4N}$: $k_1^{I} \ldots k_N^{I}$; ... ; $k_1^{IV} \ldots k_N^{IV}$

$R_{S+4N} \ldots R_{S-1+5N}$: $l_1^{I} \ldots l_N^{I}$; ... ; $l_1^{IV} \ldots l_N^{IV}$ (Zwischensummen der k)

Die Speicher $R_{ØØ}$ bis $R_{Ø8}$ sind fest belegt, während die übrigen Speicher, in Anpassung an N und an die Komplexität des Funktionensystems der f_i dynamisch zugewiesen werden können, beginnend mit R_S und endend mit R_{S-1+5N}; S wird in $R_{Ø5}$ festgelegt.

Die Funktionen f_i im Unterprogramm "FN" sind mit den Argumentwerten aus den Speichern $R_{Ø3}$ und $R_{S+2N} \ldots R_{S-1+3N}$ zu bilden und dann nach $R_{S+N} \ldots R_{S-1+2N}$ abzuspeichern.

Beispiel:

Im folgenden wird die Differentialgleichung $y'' = -y$ mit den Anfangsbedingungen $y(Ø) = Ø$; $y'(Ø) = 1$ zugrundegelegt, deren Lösung bekanntlich $y(x) = \sin x$ (x in Radian) lautet.

Durch $y_1 = y$ und $y_2 = y'$ wird hieraus ein System von 2 Differentialgleichungen 1. Ordnung:

$$y_1' = y_2 \qquad y_1(Ø) = Ø$$

$$y_2' = -y_1 \qquad y_2(Ø) = 1$$

Der Einfachheit halber soll $R_{Ø9}$ das erste Register des dynamischen Speicherbereiches sein, also S = 9. Wegen N = 2 lautet dann das Funktionsunterprogramm für f_1 und f_2 mit dem Namen ABLEIT:

```
LBLᵀABLEIT        CHS
RCL 14            STO 12
STO 11            END
RCL 13
```

Das Hauptprogramm HAUPT geht von folgender Speicherbelegung aus:

$R_{Ø4}$: N = 2 $R_{Ø5}$: S = 9 $R_{Ø6}$: h = Ø.Ø1

$R_{Ø7}$: ABLEIT $R_{Ø8}$: $x^{(Ø)} = Ø$ $R_{Ø9}$: $y_1(Ø) = Ø$

$R_{1Ø}$: $y_2(Ø) = 1$

Weiterhin wird von HAUPT verlangt, daß nach jeweils 1Ø Integrationsschritten x und y_1 zur Anzeige gebracht werden. Für diesen Schrittzähler wird der hinter dem hier definierten dynamischen Speicherbereich liegende Speicher R_{19} benutzt.

Das Hauptprogramm hat dann folgende Gestalt:

```
LBLᵀHAUPT         ARCL Ø9
1Ø                AVIEW
STO 19            LBL Ø1
CLA               XEQᵀRUKUN
FIX 1             DSE 19
ARCL Ø8           GTO Ø1
⊦Ø                GTOᵀHAUPT
FIX 6             END
```

(Ø bedeutet SPACE)

Nach der Speicherfestlegung durch XEQ SIZE Ø2Ø wird das Programm durch XEQ HAUPT gestartet. Ein Vergleich der -nach jeweils etwa 17Ø s Rechenzeit- erhaltenen Werte von x und $y_1(x)$ mit den im RAD-Modus direkt errechneten Werten von $y_1 = \sin x$ zeigt, daß das Programm fehlerlos arbeitet.

Die folgenden Seiten enthalten die Struktogramme von RUKUN und seinen Unterprogrammen nach Nassi-Shneiderman sowie das vollständige Listing.

Hamburg 73, den 23.Ø3.198Ø   K. W. Hoenow
Alaskaweg 18

Unterprogramm RUKUN

Speicherbelegung bei Programmstart:

$R_{Ø4}$: N $R_{Ø6}$: h $R_{Ø8}$: $x^{(Ø)}$

$R_{Ø5}$: S $R_{Ø7}$: "FN" R_{S-1+i}: $y_i^{(Ø)}$ i=1...N

S-1 → $R_{Ø2}$
$x=x^{(Ø)}$ → $R_{Ø3}$
Flag ØØ setzen

(UP Ø2)

Flag ØØ löschen

(UP "FN")

Flags Ø2 und Ø4 setzen

(UP Ø5)

Flag Ø2 löschen
$x=x^{(Ø)}+\frac{h}{2}$ → $R_{Ø3}$
Flag Ø1 setzen

(UP Ø2)

(UP "FN")

Flag Ø3 setzen

(UP Ø5)

(UP Ø2)

(UP "FN")

(UP Ø5)

$x=x^{(Ø)}+h$ → $R_{Ø3}$; $x^{(1)}=x$ → $R_{Ø8}$
Flag Ø1 löschen

(UP Ø2)

(UP "FN")

Flags Ø3 und Ø4 löschen

(UP Ø5)

Speicherbelegung bei Programmende:

wie bei Start, jedoch $R_{Ø8}$: $x^{(1)}$

R_{S-1+i}: $y_i^{(1)}$ i=1...N

Unterprogramm Ø2

Speicherbelegung bei Programmstart:

$R_{Ø4}$: N

R_{S-1+i} : $y_i^{(Ø)}$ i=1...N

$R_{S-1+3N+i}$: k_i i=1...N falls Flag ØØ gelöscht

i=N, 1, -1

- $y_i=y_i^{(Ø)} \rightarrow R_{S-1+2N+i}$
- Flag ØØ gesetzt?
  - ja: —
  - nein:
    - $y_i=y_i+k_i \rightarrow R_{S-1+2N+i}$
    - Flag Ø1 gesetzt?
      - ja: $y_i=y_i+k_i \rightarrow R_{S-1+2N+i}$
      - nein: —

Unterprogramm "FN"

Speicherbelegung bei Programmstart:

$R_{Ø3}$: x $R_{S-1+2N+i}$: y_i i=1...N

$y_i'=f_i(x, y_1, \ldots y_N) \rightarrow R_{S-1+N+i}$ i=1...N

Unterprogramm Ø5

Speicherbelegung bei Programmstart:

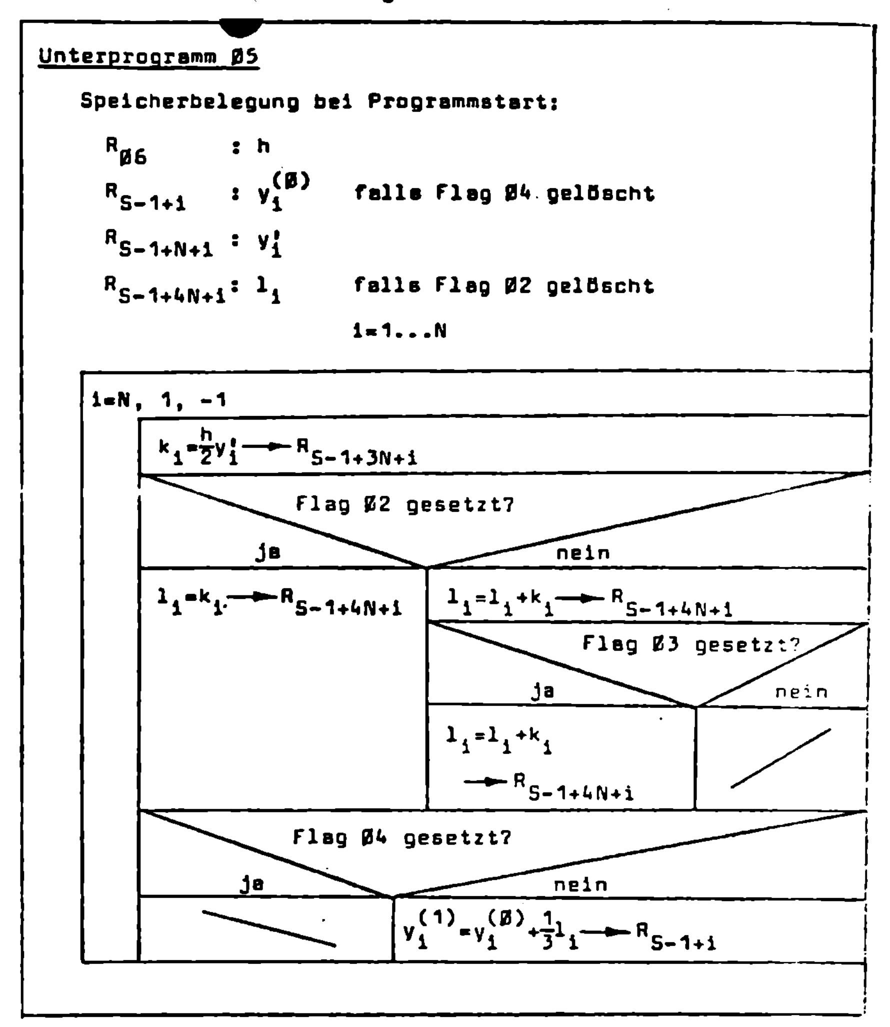
$R_{Ø6}$: h

R_{S-1+i} : $y_i^{(Ø)}$ falls Flag Ø4 gelöscht

$R_{S-1+N+i}$: y_i'

$R_{S-1+4N+i}$: l_i falls Flag Ø2 gelöscht

i=1...N



27 Register

```
    LBL"RUKUN
    RCL Ø5
    1
    -
ØØ5 STO Ø2
    RCL Ø8
    STO Ø3
    SF ØØ
    XEQ Ø2
Ø1Ø CF ØØ
    XEQ IND Ø7
    SF Ø2
    SF Ø4
    XEQ Ø5
Ø15 CF Ø2
    RCL Ø8
    RCL Ø6
    2
    /
Ø2Ø +
    STO Ø3
    SF Ø1
    XEQ Ø2
    XEQ IND Ø7
Ø25 SF Ø3
    XEQ Ø5
    XEQ Ø2
    XEQ IND Ø7
    XEQ Ø5
Ø3Ø RCL Ø8
    RCL Ø6
    +
    STO Ø3
    STO Ø8
Ø35 CF Ø1
    XEQ Ø2
    XEQ IND Ø7
    CF Ø3
    CF Ø4
Ø4Ø XEQ Ø5
    RTN
---------------------------------
    LBL Ø2        Unterprogramm
    RCL Ø4
    STO Ø1
Ø45 LBL Ø3
    RCL Ø1
    RCL Ø2
    +
    STO ØØ
Ø5Ø RCL IND ØØ
    RCL Ø4
    2
    *
    ST+ ØØ
Ø55 RDN
    STO IND ØØ
    FS? ØØ
    GTO Ø4
    RCL Ø4
Ø6Ø ST+ ØØ
    RCL IND ØØ
    RCL Ø4
    ST- ØØ
    RDN
Ø65 ST+ IND ØØ
    FC? Ø1
    ST+ IND ØØ
    LBL Ø4
    DSE Ø1
Ø7Ø GTO Ø3
    RTN
---------------------------------
    LBL Ø5        Unterprogramm
    RCL Ø4
    STO Ø1
Ø75 LBL Ø6
    RCL Ø1
    RCL Ø2
    +
    RCL Ø4
Ø8Ø +
    STO ØØ
    RCL IND ØØ
    RCL Ø6
    *
Ø85 2
    /
    RCL Ø4
    2
    *
Ø9Ø ST+ ØØ
    RDN
    STO IND ØØ
    RCL Ø4
    ST+ ØØ
Ø95 RDN
    FS? Ø2
    GTO Ø7
    ST+ IND ØØ
    FS? Ø3
1ØØ ST+ IND ØØ
    GTO Ø8
    LBL Ø7
    STO IND ØØ
    LBL Ø8
1Ø5 FS? Ø4
    GTO Ø9
    RCL IND ØØ
    3
    /
11Ø RCL Ø1
    RCL Ø2
    +
    STO ØØ
    RDN
115 ST+ IND ØØ
    LBL Ø9
    DSE Ø1
    GTO Ø6
    END
```

10 - [illegible]kte Gauß - Quadratur

Das folgende Programm benutzt die 10 Pkte Quadratur nach Gauß , um schnell und genau das Integral einer als Tastenfolge definierten Funktion zu berechnen .

Die Funktion muß als globales Label definiert werden . Bei Aufruf der Fkt ist x im X-Register vorhanden. Zur Programmierung steht der Restspeicher , der Stack und die Datenregister ab R 17 zur Verfügung . Singularitäten oder Diskontinuietäten der Funktion verursachen einen Abbruch der Integration . Diese Schwierigkeiten können mit stückweiser Integration behoben werden .

GLEICHUNGEN .

$$\int_A^B f(x)\,dx = \frac{b-a}{2} \sum_{i=1}^{10} w_i\, f\left(\frac{z_i(b-a)+b+a}{2}\right)$$

wobei :

$z_1 = .1468743390$	$z_6 = .2190863625$
$z_2 = .2955242247$	$z_7 = .8650633667$
$z_3 = .4333953941$	$z_8 = .1494513492$
$z_4 = .2692667193$	$z_9 = .9739065285$
$z_5 = .6794095683$	$z_{10} = .6667134430 \times 10^{-1}$

BEISPIEL .

$$\int_1^{10} \frac{1}{x}\,dx = \ln 10 = 2.302585093$$

Die Integration mit diesem Programm liefert : 2.302578678 (N=1)

	INPUT	TASTEN	DISPLAY
Lbl AA			
1/x			
RTN		XEQ GQ	A=?
	10	R/S	B=?
	1	R/S	N=?
	1	R/S	FUNCTION NAME?
	AA	R/S	F=2.3025

ALLGEMEINE INSTRUKTIONEN

INPUT	TASTEN	OUTPUT
1)Eingabe Fkt	Lbl --	
	......	
	......	
	RTN	
2)Anwahl Prgm	XEQ GQ	A=?
A	R/S	B=?
B	R/S	N=?
N	R/S	FUNCTION NAME?
Name	R/S	F= ----
3)Eingabe A/B	XEQ a	siehe 2)
4)Eingabe N	XEQ A	N=?
N	R/S	F= ----

weiter 1),2).,3),4).

Bemerkungen :

1) siehe Programmlinie 026 : Function Name? und ff
   Für Besitzer des MATH1A Moduls : XEQ ROM XFN

2) Programmlinien 055 , 063 , 075 X <> X erfüllt die
   Funktion einer NOP - Funktion .
   Einzugeben als : XEQ Alpha X <> Alpha . X

3) Programm eingeben ; Daten eingeben in folgende Register:

Reg Nr.	Konstante
7	z1
8	z2
9	z3
10	z4
11	z5
12	z6
13	z7
14	z8
15	z9
16	z1o

Die Daten sind mit 7,016 XEQ Alpha WDTAX Alpha
auf eine Kartenhälfte zu beschreiben .

DETLEV BOCK

0 [illegible]KTE GAUß QUADRATUR

```
 01◆LBL "GQ"
 02 1.488743
39 E-1
 03 STO 07
 04 2.955242
247 E-1
 05 STO 08
 06 4.333953
941 E-1
 07 STO 09
 08 2.692667
193 E-1
 09 STO 10
 10 6.794095
683 E-1
 11 STO 11
 12 2.190863
625 E-1
 13 STO 12
 14 8.650633
667 E-1
 15 STO 13
 16 1.494513
492 E-1
 17 STO 14
 18 9.739065
28 E-1
 19 STO 15
 20 6.667134
43 E-2
 21 STO 16
 22 CLST
 23◆LBL a
 24 "B=?"
 25 PROMPT
 26 STO 01
 27 "A=?"
 28 PROMPT
 29 STO 00
 30 SF 00
 31◆LBL A
 32 "N=?"
 33 PROMPT
 34 RCL 01
 35 RCL 00
 36 -
 37 X<>Y
 38 /
 39 2
 40 /
 41 STO 02
 42 SF 00
 43 RCL 00
 44 X<>Y
 45 -
 46 STO 03
 47 "FKT NAM
E?"
 48 AON
 49 PROMPT
 50 ASTO 05
```

```
 51 AOFF
 52◆LBL E
 53 RCL 03
 54 RCL 02
 55 +
 56 LASTX
 57 +
 58 RCL 01
 59 X<=Y?
 60 GTO e
 61 X<>Y
 62 STO 03
 63 0
 64 FS?C 00
 65 STO 04
 66 6
 67 STO 06
 68 XEQ b
 69 XEQ b
 70 XEQ b
 71 XEQ b
 72 XEQ b
 73 GTO E
 74◆LBL b
 75 ISG 06
 76 X<> X
 77 RCL 03
 78 RCL 02
 79 RCL IND
06
 80 *
 81 -
 82 XEQ IND
05
 83 ISG 06
 84 X<> X
 85 RCL IND
06
 86 *
 87 ST+ 04
 88 DSE 06
 89 RCL 03
 90 RCL 02
 91 RCL IND
06
 92 *
 93 +
 94 XEQ IND
05
 95 ISG 06
 96 X<> X
 97 RCL IND
06
 98 *
 99 ST+ 04
100 RTN
```

```
101◆LBL e
102 RCL 02
103 RCL 04
104 *
105 "F="
106 ARCL X
107 PROMPT
108 END
```

EULERSCHE GAMMAFUNKTION

Die Eulersche Gammafunktion ist in der Zahlentheorie bis zur theoretischen Physik eine oft verwendete Funktion . Sie ist durch ein uneigentliches Integral definiert :

$$\Gamma(x) = \int_0^{\infty} e^{-t}\, t^{x-1}\, dt \quad , \quad \Gamma(\tfrac{1}{2}) = \sqrt{\pi}$$

und von C.F.Gauß für alle $x \neq 0,-1,-2,...$ durch den Grenzwert

$$\Gamma(x) = \lim_{n \to \infty} \frac{n!\, n^x}{x(x+1)(x+2)\ldots(x+n)}$$

erklärt . ($n^x = e^{x \log n}$ mit positiv-reellem Logarithmus)

Bei diesem Programm wird als Näherungsberechnung für die Funktionswerte die Stirlingsche Formel benutzt :

$$\Gamma(x+1) \approx \sqrt{2\pi x}\; x^x \exp\left[\frac{-x+1}{12x}\right]$$

Für $x \geq 40$ liefert das Programm ein Ergebnis mit einem Fehler von 10^{-7} bis 10^{-8} , Für $x < 45$ wird Gamma von x mit

$$\Gamma(x) = \frac{\Gamma(z)}{(z-1)(z-2)(z-3)\ldots x}$$

wobei $z = 45 +$ Dezimalteil von x .

INSTRUKTIONEN .:

	INPUT	TASTEN	DISPLAY
1)nur beim 1.Durchlauf	x	XEQ GM	I(x)=----
2)	x	A	s.o.

Anmerkung : bei Verwendung eines Druckers kann das GAMMA-Symbol mit " 7 XEQ Alpha ACX / ACCHR Alpha " erzeugt werden .

DETLEV BOCK

```
        PRP "-"
 01◆LBL "GAM
          MA"
 02 CF 28
 03 FIX 3
 04 RAD
 05 CF 01
 06 X<=0?
 07 XEQ 00
 08 STO 01
 09 FRC
 10 X=0?
 11 GTO 02
 12 45
 13 RCL 01
 14 X>Y?
 15 GTO 09
 16 RDN
 17 +
 18 STO 02
 19 R↑
 20 -
 21 STO 04
 22 RCL 02
 23 XEQ 09
 24 RCL 02

 25◆LBL 01
 26 1
 27 -
 28 /
 29 LASTX
 30 DSE 04
 31 GTO 01
 32 FC? 01
 33 GTO 03
 34 *
 35 PI
 36 /
 37 RCL 01
 38 LASTX
 39 *
 40 SIN
 41 *
 42 1/X
 43 CHS
 44 GTO 08

 45◆LBL 09
 46 1
 47 -
 48 STO 03
 49 LN
 50 LASTX
 51 *
 52 LASTX
 53 -
 54 LASTX
 55 12
 56 *
 57 1/X
 58 +
 59 E↑X
 60 RCL 03
 61 PI
 62 *
 63 2
 64 *
 65 SQRT
 66 *
 67 RTN

 68◆LBL 00
 69 SF 01
 70 CHS
 71 RTN

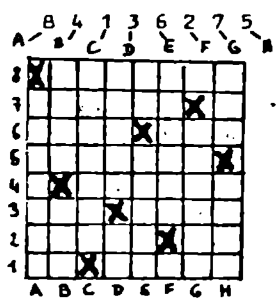
 72◆LBL 03
 73 RDN
 74 GTO 08

 75◆LBL 02
 76 LASTX
 77 1
 78 -
 79 FS? 01
 80 1 E2
 81 FACT

 82◆LBL 08
 83 "I(X)="
 84 ARCL X
 85 AVIEW
 86 .END.
```

ACHT DAMEN PROBLEM

In CHIP wurden schon öfters Programme zur Lösung der 8 Damen Problems vorgestellt,aber noch nicht für den HP41 c Diese Tatsache hat mich veranlaßt,ihnen ein solches Programm zuzusenden.
Bei diesem Spiel handelt es sich darum,acht Damen auf einem Schachbrett so aufzustellen,daß keine die andere schlagen kann.
Das Programm wird mit XEQ^{T}DAME gestartet.Es werden alle zulässigen Konfigurationen ausgegeben(92),dann gibt der Rechner einBeep und schaltet ab.Um sich die Lösungen noch einmal anzusehen schaltet man den HP-41c wieder ein.
Er bringt dann "DIE LOESUNGEN:" ins Display und zeigt nacheinander,begleitet von einem 4er-Ton,die möglichen Aufstellungen an.
Beispiel:



UM ALLE LÖSUNGEN IN DEN DATENREGISTERN UNTERZUBRINGEN, BENÖTIGT MAN ZWEI MEMORY MODULE!
Ausweg: Man läßt die fünf Befehle (PSE PSE STO IND 14 1 ST+ 14) vor Label Ø7 weg und fügt stattdessen einen STOP-Befehl ein.Allerdings ist dann die Anzeige der Lösungen nach dem OFF nicht mehr möglich.

Andreas Weiler

Listingliegt bei.

Listing:

```
*LBLTDAME
 CLRG
 FIX Ø
 CF 29
 15
 STO 14
 8
 STO ØØ
*LBL ØØ
 8
 STO IND ØØ
 DSE ØØ
 GTO ØØ
 7
 STO Ø9
*LBL Ø1
 RCL IND Ø9
 X=Ø?
 GTO 1Ø
 8
 STO 1Ø
*LBL Ø2
 RCL 1Ø
 RCL Ø9
 -
 x=Ø?
 GTO Ø4
 STO 11
 STO 12
*LBL Ø3
 RCL IND 1Ø
 RCL [illegible] 11 -
 +
 STO 13
 RCL IND Ø9
 -
 x=Ø?
 GTO Ø7
 RCL 11
 RCL 12
 +
 x≠Ø?
 GTO Ø9
 RCL Ø9
 RCL 1Ø
 -
 CHS
 1
 -
 x≠Ø?
 GTO Ø8
 RCL Ø9
 1
 -
 x=Ø?
 GTO Ø5
*LBL Ø4
 1
 ST- Ø9
 GTO Ø1
```

```
 LBL Ø5
 8
 STO ØØ
 CLST
*LBL Ø6
 RCL IND ØØ
 RCL ØØ
 1
 -
 1Ø^x
 x
 +
 DSE ØØ
 GTO Ø6
 TONE 8
 VIEW X
 PSE        Stop
 PSE
 STO IND 14
 1
 ST+ 14
*LBL Ø7
 1
 ST- IND Ø9
 GTO Ø1
*LBL Ø8
 1
 ST- 1Ø
 GTO Ø2
*LBL Ø9
 RCL 12
 ST- 11
 GTO Ø3
*LBL 1Ø
 RCL Ø9
 8
 -
 x=Ø
 GTO 11
 8
 STO IND Ø9
 1
 ST+ Ø9
 GTO Ø7
*LBL 11
 BEEP
 BEEP
 SF 11
 OFF
 FC? 55
 GTO 13
 TDIE LOESUNGEN:
 AVIEW
 PSE
 BEEP
 15.1Ø6
 STO L
*LBL 12
 RCL IND L
 TONE 4
 PSE
```

```
 ISG L
 GTO 12
*LBL 13
 CLST
 FIX 4
 BEEP
 END.
```

Speicherplan:

ØØ: Arbeitsspeicher
Ø1:r1
Ø2:r2
Ø3:r3
Ø4:r4
Ø5:r5
Ø6:r6
Ø7:r7
Ø8:r8
Ø9:z
1Ø:u
11:j
12:u-z
13:i
14:Zähler
15:Lösung 1
16-1o6:Losung 2-92

Flags:

CF 29
FC? 55
SF 11

Andreas Weiler

21. 3.1980 Kropf Walter, Schönowitzstraße 14, .. 37 Leoben

Diskrete Fourier-Transformation

Programmzeilen: 249 444Bytes 64Register
Datenspeicher: (K+1).2+T+6
System Konfiguration: 1 Memorymodul, Kartenleser, (Printer)

Das Programm berechnet aus einer Anzahl von Abtastwerten einer harmonischen Funktion die Frequenzkomponenten. Die Anzahl der Abtastwerte ist nur durch den Speicherraum begrenzt, die Anzahl der Frequenzkomponenten ist beliebig.Die Abtastwerte werden am Beginn fortlaufend eingegeben, die Durchführung der Transformation erfolg dann automatisch. Das Resultat kann in rechtwinkliger oder polarer Form ausgegeben werden. Bei polarer Ausgabe können die Oberwellen in % der Grundwelle ausgegeben werden. Das Programm ist zur Verwendung mit oder ohne Drucker geeignet.

Wenn eine Berechnung abgeschlossen ist, können weitere Frequenzfolgen ohne neue Eingabe angehängt werden. Für spätere Verwendung können die Eingabewerte auf Magnetkarte gespeichert werden.

Das Eingabeformat kann durch Änderung des Programmschrittes 29, das Ausgabeformat durch Änderung der Schritte 158, 159, 161 bzw. 168 leicht den Erfordernissen der Genauigkeit angepaßt werden.

Die Anzahl der Abtastwerte soll mindestens doppelt so groß sein als die Ordnungszahl der höchsten zu berechnenden Frequenz.

Grenzen und Einschränkungen:
Wenn die Abtastwerte von einer Magnetkarte eingelesen werden, muß die Anzahl der auf der Karte gespeicherten Abtastwerte und die Eingabe T=... übereinstimmen.
Der Abstand der Abtastwerte muß konstant sein.
Wenn kein Drucker angeschlossen ist, müssen aufeinanderfolgende Ausgabewerte mit R/S abgerufen werden.

Verwndete Formeln:

T=Periodendauer als Summe der Abtastwerte t
K=Anzahl der zu berechnenden Frequenzen k
kl= erste Oberwelle welche berechnet wird

$$f(t) = \frac{a_0}{2} + \sum_1^k a_k \cdot \cos k.2.\pi.t/T + b_k \cdot \sin k.2.\pi.t/T = \frac{a_0}{2} + \sum_1^k c_k \cdot e^{-j \cdot \varphi_k}$$

$$a_k = \frac{2}{T} \cdot \sum_1^T f(t).\cos \frac{k.2.\pi.t}{T} \qquad b_k = \frac{2}{T} \cdot \sum_1^T f(t).\sin \frac{k.2.\pi.t}{T}$$

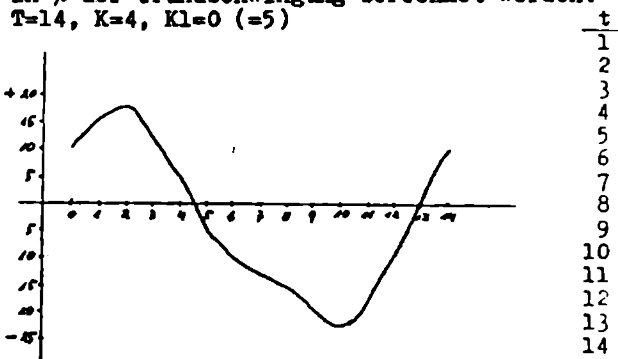
An alle Clubmitglieder:

Oliver Rietschel
Hewlett-Packard Anwender-Club
Postscheckkonto Hmb. 48635-207
Postfach 373 · D-2420 Eutin

Beispiel: Eine periodische Funktion hat den unten abgebildeten Verlauf. Es sind die Harmonischen bis zur 8. Oberwelle zu bestimmen. Es stehen 30 Datenregister zur Verfügung. Bei einer Transformation können

$$K=(30-6-14):2-1 = 4 \text{ Frequenzen}$$

berechnet werden. Die Transformation muß in zwei Rechengängen durchgeführt werden.
Das Ergebnis soll rechtwinklig und polar ausgegeben werden. Bei der polaren Ausgabe sollen die Oberwellen in % der Grundschwingung berechnet werden.
T=14, K=4, K1=0 (=5)



t	f(t)
1	16
2	18
3	12
4	5
5	-4,5
6	-10
7	-13
8	-15
9	-20
10	-22
11	-18
12	-10
13	0
14	10

Beispiel: Lösung

Eingabe	Taste		Kommentar
	A	T=14	
14	R/S	1= 16,000	Eingabe von T
16	R/S	2= 18,000	
18	R/S	3= 12,000	Eingabe der Abtastwerte
		4= 5,000	
		5= -4,500	
		6= -10,000	
		7= -13,000	
		8= -15,000	
		9= -20,000	
		10= -22,000	
		11= -18,000	
		12= -10,000	
		13= 0,000	
		14= 10,000	
4	R/S	K=4	Anzahl der Frequenzen
0	R/S	K1=0	1.Frequenz
		END DFT	Berechnung
	D	K : b : a	Ausgabe rechtwinklig
		0: 0,000: -7,357	
		1: 14,099: 11,807	
		2: 3,155: 2,255	
		3: -0,745: -0,574	
		4: 0,422: -0,067	
	E	K : PHI : c	Ausgabe polar
		0: 3,142: 7,357	
		1: 0,874: 18,390	nach K=1 mit R/S stoppen
	f e	K : PHI : c	
18,390	R/S	0: 3,142: 7,357: 20,0%	Eingabe des Wertes für 100%
		1: 0,874: 18,390: 100,0%	
		2: 0,950: 3,878: 21,1%	
		3: 4,056: 0,941: 5,1%	
		4: 1,729: 0,428: 2,3%	
	C		Start für nächste Frequenzgruppe
3		K=3	
5		K1=5	
		END DFT	
	D	K : b : a	
		5: -0,180: 0,017	
		6: -0,296: -0,009	
		7: 0,000: 0,500	
		8: 0,296: -0,009	
	f e		
18,390	R/S	K : PHI : c	
		5: 4,804: 0,181: 1,0%	
		6: 4,683: 0,297: 1,6%	
		7: 0,000: 0,500: 2,7%	
		8: 1,600: 0,297: 1,6%	

Programmablauf: Karte-Input-Karte START DFT

Nr.	Anweisung	Eingabe	Taste	Anzeige
1	Programmstart		A	"T="
2	Anzahl der Abtastwerte gleichen Abstands	T	R/S	"1 "
3	Amplitude des 1.Abtastwertes	f(t)	R/S	"2 "
	Schritt 3 wiederholen für alle Abtastwerte, am Schluß			"K="
4	Anzahl der zu berechnenden Frequenzen	K	R/S	"K1="
5	Ordnungszahl der 1. zu berechnenden Frequenz	K1	R/S	"1"
	während der Rechenzeit wird die fortschreitende Abarbeitung der Abtastwerte im Display angezeigt. Ende der Berechnung			BEEP "END DFT"
6	Wenn die Abtastwerte für später auf Datenkarte gespeichert werden		f b	"WDAT INP"
7	Wenn Programm mit Daten von Karte gestartet werden soll		f a	"T="
	Anzahl der Abtastwerte	T	R/S	"RDAT INP"
	Datenkarte einlesen			"K="
	weiter bei Schritt 4 Wenn T nicht mit der Anzahl der gespeicherten Abtastwerte übereinstimmt, wird Schritt 7 wiederholt.			
8	Korrektur falscher Eingaben:		B	ignor.
	richtige Werte eingeben	t	ENTER	
		f(t)	R/S	
	weiter bei Schritt 3			
	oder, wenn die restlichen Werte für t bereits eingegeben waren		C	"K="
	weiter bei Schritt 4			
9	Ausgabe der Koeffizienten a, b,		D	K:b:a
10	Ausgabe der Koeffizienten c, φ ,		E	K:Phi:c
11	Ausgabe von c, φ, mit % Anzeige: Wert von cl nach Schritt 10 berechnen, Prgr.stoppen		R/S f e	"cl="
	cl für Grundwelle eingeben	cl	R/S	K:Phi:c:%
12	für Berechnung einer weiteren Folge von Frequenzen		C	"K="
	weiter bei Schritt 4			
	für neue Rechnung nach Schritt 1 oder 7			

Status: Size (K+1).2+T+6, Prgr.Reg. 64, User Mode ON

Flags: 00 Pointer t setzen
01 Eingabe von Karte, Resultat polar
02 Ausgabe mit %
03 negativ, so %

Tastenzuordnung: FOUR : 11

Register: 00 Pointer t, k
01 Pointer k
02 K
03 kl
04 t, cl/100
05 T
06 t Eingabewerte
xx "
xx a Koeffizienten
xx b "
xx a "
xx b "

PROGRAM LISTING

PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

MA 102

Line	Keystrokes	Comments
01	♦LBL "FOUR"	Programmstart
02	CF 01	
03	♦LBL 10	
04	SF 21	
05	RAD	
06	CF 29	
07	FIX 0	
08	"T="	T speichern
09	XEQ 09	
10	STO 05	
11	FS?C 01	Rücksprung wenn
12	RTN	Eingabe von Karte
13	1	
14	STO 04	Zähler für tl
15	FC? 55	
16	CF 21	
17	♦LBL B	Start Dateneingabe
18	" "	
19	FIX 0	
20	RCL 04	ti anzeigen
21	ARCL X	
22	PROMPT	f(ti) eingeben
23	" "	
24	ARCL Y	
25	"⊢= "	
26	FIX 3	Eingabe Format
27	ARCL X	
28	AVIEW	Eingabe drucken
29	X<>Y	
30	STO 04	
31	5	
32	+	
33	X<>Y	
34	STO IND Y	
35	1	ti+1
36	ST+ 04	
37	RCL 05	
38	RCL 04	
39	X<=Y?	alle Stützwerte
40	GTO B	eingegeben?
41	♦LBL 11	
42	SF 21	
43	♦LBL C	Start Fouriertransf.
44	FIX 0	
45	"K="	K speichern
46	XEQ 09	
47	STO 02	
48	"K1="	KI speichern
49	XEQ 09	
50	STO 03	
51	SF 00	
52	XEQ 00	Pointer setzen k. t
53	CLX	
54	♦LBL 01	

Line	Key pressed	Comments
55	STO IND 01	Speicherbereich
56	ISG 01	k löschen
57	GTO 01	
58	1	
59	STO 04	
60	CF 21	
61	♦LBL 02	
62	CLA	Schleife DFT
63	ARCL 04	
64	AVIEW	t in Anzeige
65	XEQ 00	Pointer setzen k
66	-1	
67	STO 02	Zähler für k
68	♦LBL 03	Schleife
69	1	
70	ST+ 02	K+1
71	RCL 02	
72	RCL 03	
73	+	
74	2	
75	*	
76	PI	k.2. .t/T
77	*	
78	RCL 04	
79	*	
80	RCL 05	
81	/	
82	RCL IND 00	f(t)
83	P-R	
84	ST+ IND 01	f(t).cos
85	X<>Y	
86	ISG 01	
87	ST+ IND 01	f(t).sin
88	ISG 01	nächstes k
89	GTO 03	
90	1	t+1
91	ST+ 04	
92	ISG 00	nächstes t
93	GTO 02	
94	XEQ 00	Pointer setzen k
95	2	
96	RCL 05	2/T
97	/	
98	♦LBL 04	
99	ST* IND 01	Koeff. .2/T
100	ISG 01	
101	GTO 04	
102	SF 21	
103	BEEP	
104	"END DFT"	
105	AVIEW	
106	PROMPT	
107	♦LBL 00	
108	RCL 05	

LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

Keystrokes / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commento
109 5	
110 +	
111 1 E3	
112 /	
113 6	
114 +	
115 FS?C 00	
116 STO 00	Pointer setzen t
117 RCL 02	
118 1	
119 +	
120 500	
121 /	
122 +	
123 RCL 05	
124 +	
125 STO 01	Pointer setzen k
126 RTN	
127•LBL e	Resultatausgabe
128 CLX	mit %
129 "c1="	Wert für 100%
130 PROMPT	speichern
131 100	
132 /	
133 STO 04	
134 SF 02	
135•LBL E	Resultatausgabe
136 "K : PHI : c"	polar
137 AVIEW	
138 SF 01	
139 GTO 05	
140•LBL D	Resultatausgabe
141 "K b a"	rechtwinklig
142 AVIEW	
143 CF 01	
144•LBL 05	Pointer setzen k
145 XEQ 00	
146 RCL 03	kl- ROO
147 STO 00	Schleife
148•LBL 06	
149 FIX 0	
150 CLA	ki in Anzeige
151 ARCL 00	
152 "⊢: "	
153 RCL IND 01	
154 ISG 01	
155 RCL IND 01	
156 FS? 01	
157 XEQ 08	Ausgabeformat
158 FIX 3	
159 RND	
160 X<>Y	
161 RND	
162 ARCL Y	b(φ) in Anzeige

Line / Zeile / Ligne / Linea — Key pressed / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commenti
163 "⊢: "	
164 ARCL X	a(c) in Anzeige
165 FC? 02	
166 GTO 07	
167 "⊢: "	
168 FIX 1	% Format
169 R↑	
170 RND	% in Anzeige
171 ARCL X	
172 "⊢%"	
173•LBL 07	
174 AVIEW	Resultat anzeigen
175 1	
176 ST+ 00	ki+1
177 ISG 01	
178 GTO 06	nächstes k
179 CF 01	
180 CF 02	
181 BEEP	
182 ADV	
183 RTN	
184•LBL 08	Resultat polar
185 X<>Y	
186 R-P	
187 X<>Y	
188 X<0?	
189 SF 03	
190 2	
191 PI	
192 *	
193 FC?C 03	$\varphi < 0$: +2π
194 CLX	
195 +	
196 FC? 02	
197 RTN	
198 X<>Y	
199 ENTER↑	
200 ENTER↑	
201 RCL 00	k=0: ao/2
202 X=0?	
203 SF 03	
204 RDN	
205 FS? 03	
206 2	
207 FS?C 03	
209 /	
209 RCL 04	
210 /	
211 RDN	
212 X<>Y	
213 RTN	
214•LBL 09	
215 PROMPT	
216 ARCL X	

Line / Zeile / Ligne / Linea	Keystrokes / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commenti
217	FS? 55	
218	PRA	Eingabe drucken
219	RTN	
220	◆LBL b	
221	SF 00	
222	XEQ 00	Pointer stzen t.
223	RCL 00	
224	1	
225	-	
226	CF 21	
227	"WDAT INP"	
228	AVIEW	Eingabedaten auf
229	WDTAX	Karte
230	GTO 11	
231	◆LBL a	
232	SF 01	
233	XEQ 10	
234	SF 00	
235	XEQ 00	Pointer setzen t
236	RCL 05	
237	RCL 00	
238	1	
239	-	
240	CF 21	
241	"RDAT INP"	
242	AVIEW	Eingabedaten lesen
243	RDTAX	
244	RDN	
245	RCL 05	
246	X≠Y?	T falsch: Eingabe
247	GTO a	wiederholen
248	GTO 11	
249	END	
	LBL"FOUR	
	END 444 BYTES	

24. 3.1980 Kropf Walter, Schönowitzstraße 14, A-[illegible].00 [illegible]ben

Inverse Fourier-Transformation

Programmzeilen: 199 398Bytes 57Register
Datenspeicher: n.3+17
System Konfiguration: 1 Memorymodul, Kartenleser, (Printer)

Das Programm berechnet aus einer Anzahl von Frequenzkomponenten die harmonische Funktion. Die Fourierkoeffizienten können in polarer oder rechtwinkliger Form vorliegen. Bei Verwendung des Printers kann die Summenfunktion auch als Diagramm dargestellt werden.
Das Programm stellt die Ergänzung zum Programm "Diskrete Fourier-Transformation" dar.
Das Programm erfordert nicht die Eingabe einer geschlossenen Folge von Frequenzen. Es brauchen nur signifikante Oberschwingungen eingegeben werden.
Das Ausgabeformat entspricht dem, vor Abruf der Berechnung gespeicherten Format.
Bei Darstellung der Funktion durch den Printer:

- Name: FT
- X-Achse: Zeit
- Y-Achse: f(t)

Grenzen und Einschränkungen:
Der Phasenwinkel Phi muß im Winkelmaß Radiant eingegeben werden.

Verwendete Formeln:
T=Periodendauer (frei wählbar)
ak, bk, ck, k,=Fourierkoeffizienten der Ordnungszahl k
n= Anzahl der gegebenen Frequenzkomponenten

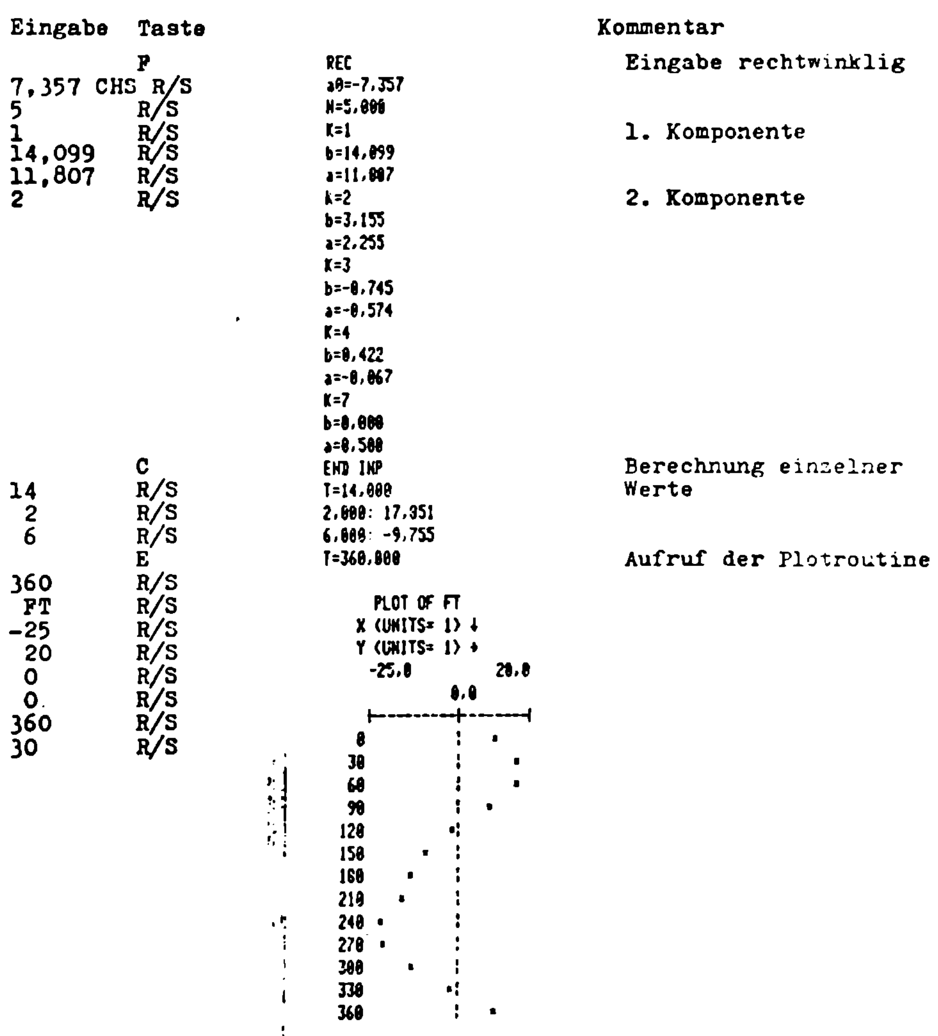
An alle Clubmitglieder:

Oliver Rietschel
Hewlett-Packard Anwender-Club
Postscheckkonto Hmb. 48635-207
Postfach 373 · D-2420 Eutin

Beispiel: Eine harmonische Schwingung besteht aus folgenden Komponenten:

k	b	a	
			ao= -7,357
1	14,099	11,807	
2	3,155	2,255	
3	-0,745	-0,574	
4	0,422	-0,067	
7	0,0	0,5	

Es soll die Funktion für die Werte t=2 und t=6 bei einer Periodenlänge von T=14 berechnet werden. DerVerlauf der Funktion ist graphisch darzustellen, wobei die Periodenlänge mit 360 bezeichnet wird.



Eingabe	Taste		Kommentar
	F	REC	Eingabe rechtwinklig
7,357 CHS	R/S	a0=-7,357	
5	R/S	N=5,000	
1	R/S	K=1	1. Komponente
14,099	R/S	b=14,099	
11,807	R/S	a=11,807	
2	R/S	k=2	2. Komponente
		b=3,155	
		a=2,255	
		K=3	
		b=-0,745	
		a=-0,574	
		K=4	
		b=0,422	
		a=-0,067	
		K=7	
		b=0,000	
		a=0,500	
	C	END INP	Berechnung einzelner Werte
14	R/S	T=14,000	
2	R/S	2,000: 17,351	
6	R/S	6,000: -9,755	
	E	T=360,000	Aufruf der Plotroutine
360	R/S		
FT	R/S	PLOT OF FT	
-25	R/S	X (UNITS= 1) ↓	
20	R/S	Y (UNITS= 1) →	
0	R/S	-25,0 20,0	
0	R/S	0,0	
360	R/S		
30	R/S		

Programmablauf:

Nr.	Anweisung	Eingabe	Taste	Anzeige
1	Programmstart		F	'REC'
2	Wenn die Eingabe polar erf.soll		f c	'POL'
	Taste f c dient als Schalter		R/S	'a0='
3	Eingabe von a0	a0	R/S	'N='
4	Anzahl der gegebenen Frequ.	n	R/S	'K='
5	Ordnungszahl der 1.Frequenz	k	R/S	'b=' ('PHI=')
	Faktor bk (φk in RAD)	bk (φk)	R/S	'a=' ('c=')
	Faktor ak (ck)	ak (ck)	R/S	'K='
	Schritt 5 wiederholen für alle Frequenzkomponenten			BEEP 'END INPUT'
6	Korrektur einer falschen Eingabe		B	'K='
	falschen Wert k eingeben	k	R/S	'K='
	nach Schritt 5 und richtige Werte eingeben			
7	Wenn die Eingabewerte auf Karte gespeichert werden sollen		f b	'WDAT INP'
8	Programmstart mit Daten von Karte		f a	'REC'
	nach Schritt 2, 4			'RDAT INP'
	Datenkarte einlesen			BEEP 'END INPUT'
	Der Wert für n muß mit den Werten auf der Karte übereinstimmen, sonst wird Schritt 8 wiederholt.			
9	Berechnung von f(t) für einzelne Werte		C	'T='
10	Periodendauer eingeben	T	R/S	'T1='
11	t eingeben	t	R/S	t:f(t)
	Schritt 11 für beliebige T wiederholen			
12	Berechnung von f(t) für eine Folge t		D	'T='
13	Periodendauer eingeben	T	R/S	'T1='
14	ersten Wert für t eingeben	t1	R/S	'dT='
15	Schrittweite dt eingeben	dt	R/S	t:f(t)
	Wenn kein Drucker angeschlossen ist, müssen folgende Ergebnisse mit R/S abgerufen werden.			
16	graphische Darstellung der Funktion		E	'T='
17	Periodendauer eingeben	T	R/S	'NAME'
	Werte für Plotroutine eingeben Name FT, Y=f(t), X=t			

Status: Size n.3+17, Prgr.Reg. 57, User Mode ON

Flags: 00 Koeff.polar
01 Input von Karte, Einzelschritt
02 Plotroutine

Tastenzuordnung: INFOUR : 21

Register: 00 Plotregister　　08 t
11 "　　10 dt
12 T
13 (n.3+16)/EEX3+17
14 Pointer
15 (R13)-2, f(t)　　ab R15 werden die Register auf Datenkarte aufgezeichnet
16 a0
17 k
18 bk (k)
19 ak (ck)
20 k
"　　Speicherlänge variabel nach Anzahl der Glieder
"

PROGRAM LISTING

PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

Line / Zeile / Ligne / Linea — Keystrokes / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commenti	Line / Zeile / Ligne / Linea — Key pressed / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commenti
01♦LBL "INFOUR"	Programmstart	55 XEQ 02	
02 CF 01		56 STO IND 14	
03♦LBL 09		57 ISG 14	Eingabe beendet?
04 SF 21		58 GTO 01	
05 RAD		59♦LBL 00	
06 CF 29		60 SF 21	
07 SF 00		61 BEEP	
08♦LBL c		62 "END INP"	
09 FS? 00	Koeffizienten polar oder	63 AVIEW	
10 "REC"	rechtwinklig ?	64 RTN	
11 FS?C 00		65♦LBL 02	
12 GTO 06		66 PROMPT	Eingabe anfordern
13 SF 00		67 ARCL X	
14 "POL"		68 FS? 55	
15♦LBL 06		69 PRX	
16 AVIEW		70 RTN	
17 FIX 3		71♦LBL B	Eingabekorrektur
18 "a0="		72 RCL 13	Pointer setzen
19 FC? 01	a_0 speichern	73 STO 14	
20 XEQ 02		74 "K="	
21 FC? 01		75 PROMPT	
22 STO 16		76 FIX 0	
23 "N="		77 ARCL X	
24 XEQ 02	n eingeben	78 "⊢ KORR."	
25 3		79 FS? 55	
26 *	Pointer berechnen	80 PRA	
27 16		81♦LBL 05	
28 +		82 RCL IND 14	k suchen
29 RCL IND X	Speicherbereich prüfen	83 X=Y?	
30 RDN		84 GTO 01	
31 1 E3		85 RDN	
32 /		86 ISG 14	
33 17		87 ISG 14	
34 +		88 ISG 14	
35 STO 13		89 GTO 05	
36 STO 14	Pointer setzen	90 GTO J	
37 FS?C 01	Rücksprung, wenn Eingabe	91♦LBL b	
38 RTN	von Karte	92 RCL 13	Eingabedaten auf
39♦LBL 01		93 2	Karte schreiben
40 FIX 0		94 -	
41 "K="	k speichern	95 STO 15	
42 XEQ 02		96 CF 21	
43 STO IND 14		97 "WDAT INP"	
44 ISG 14		98 AVIEW	
45 FIX 3		99 WDTAX	
46 "b="	b_k (φ_k) speichern	100 GTO 00	
47 FS? 00		101♦LBL a	Eingabedaten von
48 "PHI="		102 SF 01	Karte lesen
49 XEQ 02		103 XEQ 09	
50 STO IND 14		104 2	
51 ISG 14		105 -	
52 "a="	a_k (c_k) speichern	106 CF 21	
53 FS? 00		107 "RDAT INP"	
54 "c="		108 AVIEW	

Please use paper glue to attach listings. Adhesive tape may affect print!
Bitte Listings mit Papierleim einkleben. Klebefilme können Druck bleichen!

S.V.P. utilisez de la colle à papier pour fixer les listings. Les rubans adhésifs peuvent altérer l'impression!
Per favore usare la colla per fissare i listati. Il nastro adesivo può alterare la stampa!

PROGRAM LISTING

PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

Line / Zeile / Ligne / Linea	Keystrokes / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commento
109	RDTAX	
110	RCL 15	
111	X≠Y?	n prüfen
112	GTO a	
113	GTO 00	
114	♦LBL E	Funktion graphisch darstellen
115	SF 02	
116	XEQ 07	
117	CF 02	
118	BEEP	
119	RTN	
120	♦LBL C	einzelne Werte berechnen
121	SF 01	
122	CF 02	
123	GTO 07	
124	♦LBL D	eine Folge von Werten berechnen
125	CF 01	
126	CF 02	
127	♦LBL 07	
128	SF 21	
129	"T="	T speichern
130	XEQ 02	
131	STO 12	
132	FS? 02	
133	GTO "PRPLOT"	
134	♦LBL 04	
135	"T1="	t1 speichern
136	PROMPT	
137	STO 08	
138	FS? 01	
139	GTO 03	
140	"dT="	ΔT speichern
141	XEQ 02	
142	STO 10	
143	RCL 08	
144	♦LBL "FT"	
145	♦LBL 03	Berechnung von f(t)
146	STO 06	
147	RCL 13	Pointer setzen
148	STO 14	
149	RCL 16	
150	2	
151	/	
152	STO 15	
153	♦LBL 00	Schleife
154	RCL IND 14	
155	RCL 08	
156	*	
157	RCL 12	
158	/	
159	2	
160	*	
161	PI	k.2.π.t/T
162	*	

Line	Key pressed	Comments
163	ISG 14	
164	RCL IND 14	b (φ)
165	ISG 14	
166	RCL IND 14	a (c)
167	FS? 00	
168	P-R	
169	RCL Z	
170	1	
171	P-R	
172	ST* Z	
173	RDN	
174	ST* Z	
175	RDN	
176	+	
177	ST+ 15	$a_k \cdot \cos + b_k \cdot \sin$
178	ISG 14	
179	GTO 00	
180	RCL 15	Ende der Berechnung
181	FS? 02	
182	RTN	
183	CLA	
184	ARCL 05	
185	"F: "	
186	ARCL 15	
187	AVIEW	
188	FS? 01	
189	GTO 04	
190	RCL 10	t + Δt
191	ST+ 08	
192	RCL 12	
193	RCL 08	
194	X<=Y?	
195	GTO 03	nächstes t
196	BEEP	
197	ADV	
198	RTN	
199	END	
	LBL'INFOUR	
	LBL'FT	
	END 398 BYTES	

Please use paper glue to attach listings. Adhesive tape may affect print!
Bitte Listings mit Papierleim einkleben. Klebefilme können Druck bleichen!
S.V.P. utilisez de la colle à papier pour fixer les listings. Les rubans adhésifs peuvent altérer l'impression.
Per favore usare la colla per fissare i listati. Il nastro adesivo può alterare le stampato!

SUPERHIRN HP - 41 C

Ein Programm für den programmierbaren Taschenrechner HP-41C

Autor : Ulrich Davertzhofen , 1979/80

Das Spiel

Spieler A steckt x Codestecker in die x Löcher , ohne daß der Entschlüsseler B die geheime Steckerkombination einsehen kann. Bei der Zusammenstellung des Geheimcodes ist es gestattet , auch 2 oder mehr Stecker der gleichen Farbe zu verwenden (z.B. rot,weiß,rot,grün). Spieler B versucht nun, den Code exakt nachzubilden. Nach jedem Versuch gibt A Auskunft über den Stand der Dechiffrierung durch schwarze und weiße Chiffrierstifte. Dabei bedeutet :

1 schwarzer Stift : ein Codestecker entspricht nach Position und Farbe einem Stecker der geheimen Kombination ;

1 weißer Stift : ein Codestecker entspricht nach der Farbe, aber nicht nach der Position einem Stecker der geheimen Kombination.

Steckt A als Information x schwarze Stifte, ist der Code vollständig entschlüsselt und das Spiel beendet.

Das Programm

Der Rechner entschlüsselt jeden Code, der vom menschlichen "Gegenspieler" A z.B. auf Papier notiert wird, mit bis zu 9 Positionen (x) und bis zu 9 Farben (y). Dabei wird jeder Farbe eine Ziffer (1-9) zugeordnet (z.B. 1=freie Stelle, 2=weiß, ... , 9=schwarz). Zu den einzelnen Versuchen des Rechners gibt A die entsprechenden Informationen in der Form s.w (Anzahl schwarze.Anzahl weiße Chiffrierstifte) ein, wonach der Rechner bis zur vollständigen Entschlüsselung seine(n) nächsten Versuch(e) ermittelt.

HP 41C PROGRAM SUBMITTAL FORM

PROGRAMMFORMBLATT/DOCUMENTATION DU PROGRAMME/GENERALITÀ SUL PROGRAMMA

Program Title / Programmtitel / Titre du programme / Titolo del programma: Inverse Mastermind with HP - 41 C

Category No. / Kategorie Nr. / Catégorie N° / Categoria N°: 821

Name / Rubrik / Rubrique / Nome della categoria: Board / Table Games

No. of program lines: 245

No. of data registers / Anzahl der benötigten Datenspeicher / Nombre de registres de données / N° di registri utilizzati: 110 (62 : PRG- REGISTS. / 48 : DATA- REG.)

Recommended HP 41C System configuration / Empfohlene System-Konfiguration / Configuration recommandée / Configurazione raccomandata:
Port # 1: Memory Module
Port # 2:
Port # 3:
Port # 4:

This program requires the following programs as subroutines:
Dieses Programm benutzt folgende Programme als Unterprogramme:
Ce programme utilise les programmes suivants comme sous-programmes:
Questo programma usa i seguenti programmi come subroutines:

HP Applications ROM / HP Applications ROM / ROM d'application HP / ROM di applicazione HP:

Program Name: / Programm: / Nom du programme: / Programma:

Program Abstract / Kurzbeschreibung / Résumé / Breve descrizione del programma:
HP-41 FINDS YOUR SECRET CODE WITH UP TO 9 POSITIONS AND UP TO 9 COLOURS. AFTER EACH GUESS YOU TELL THE MACHINE HOW MANY NUMBERS ARE RIGHT IN POSITION AND COLOUR (BLACK KEY-PEG) AND HOW MANY ARE RIGHT IN COLOUR ONLY.* WORKING UP THESE INFORMATION HP DECODES YOUR COMBINATION IN ABOUT 8 GUESSES (WITH 5 POSITIONS AND 9 COLOURS). THE PROGRAM RUNS FAST.

* (WHITE KEY-PEG)

Name / Name/Nom/Nome: Ulrich Davertzhofen

Address / Strasse/Adresse/Indirizzo: Huenefeldstrasse 109

City / Ort / Localité / Città: Wuppertal   **Postal Code** / Postleitzahl / Code postal / C.A.P.: 5600   **Country** / Land / Pays / Paese: W-Germany

ACKNOWLEDGMENT AND AGREEMENT

Erklärung und Ermächtigung/Déclaration et Autorisation/Dichiarazione e Autorizzazione

To the best of my knowledge, I have the right to contribute this program material without breaching any obligation concerning nondisclosure or confidential information of other persons or organizations. I am contributing this program material on a nonconfidential nonobligatory basis to Hewlett-Packard S.A. ("HP") for inclusion in its program library, and I agree that HP may use, duplicate, modify publish, and sell the program material, and authorize others to do so without obligation or liability of any kind. HP may publish my name and address, as the contributor, to facilitate user inquiries pertaining to this program material.

Ich versichere nach bestem Wissen, dass ich über meinen Programmbeitrag frei verfügen kann, ohne dass sich dadurch für HP, andere Programmierer [illegible] irgendwelche Verpflichtungen gegenüber Dritten oder sonstige rechtliche Nachteile ergeben. HP kann meinen Programmbeitrag ohne Geheimhaltungs- und sonstige Verpflichtungen in beliebiger Weise benutzen oder verwerten. Gegen eine Veröffentlichung meines Namens im Zusammenhang mit meinem Programmbeitrag habe ich nichts einzuwenden.

Au mieux de ma connaissance, je déclare avoir le droit de fournir le présent programme sans enfreindre des obligations de secret à l'égard d'autres personnes ou organismes. Je fournis le présent programme à la Société Hewlett-Packard S.A. («HP») sur une base non confidentielle, pour incorporation dans sa bibliothèque de programmes [illegible] HP, qui pourra à son tour autoriser d'autres personnes, à l'utiliser, le reproduire, le modifier, le publier et le distribuer, sans obligations ni responsabilités d'aucune sorte. HP est autorisé à porter mes nom et adresse en tant qu'auteur du présent programme, en vue de faciliter les échanges d'informations avec les utilisateurs de ce programme.

Per quanto ne so e conosciamo, ho il diritto di fornire questo programma senza violare alcun obbligo di segreto o confidenzialità verso altre persone o organizzazioni. [illegible] programma alla Hewlett-Packard S.A. («HP») su una base non confidenziale per includerlo nella sua biblioteca di programmi e autorizzo la società HP [illegible] a sua volta piena autorizzazione a questo senza altre persone, a utilizzarlo, riprodurlo, modificarlo, pubblicarlo e distribuirlo senza obbligo né responsabilità di alcuna specie. [illegible] a sua discrezione, pubblicare il mio nome e indirizzo quale autore del presente programma onde facilitare lo scambio d'informazioni con gli utilizzatori dello stesso.

Date / Datum / Date / Data: April 1980

Signature / Unterschrift / Signature / Firma: Ulrich Davertzhofen

PROGRAM DESCRIPTION I

PROGRAMMBESCHREIBUNG I
DESCRIPTION DU PROGRAMME I
DESCRIZIONE DEL PROGRAMMA I

Application, Equations, Variables
Anwendung, Gleichungen, Veränderliche
Application, Equations, Variables
Applicazione, Equazioni, Variabili

ABOUT STRATEGY :
WORKING UP THE INFORMATION THE CALCULATOR TRIES TO ARRANGE A NEW COMBINATION WHICH IS NOT IN CONTRADICTION TO HIS FORMER GUESSES.

EXAMPLE: SECRET CODE: 2616					
1.	5555	0.0	4.	6216	2.2
2.	1111	1.0		6216	
	1			621	
3.	2221	1.1		2 1	
	21			2 16	
	2 1			2616	
	2 1		5.	2616	4.0
	1			SOLVED	
	12				
	21				

Operating limits and Warnings
Grenzen und Einschränkungen
Limites et remarques
Limiti operativi e avvertenze

1) LOOK FOR CORRECT INPUT OF YOUR INFORMATION.
2) NUMBER OF POSITIONS / COLOURS ≤ 9

PROGRAMMBESCHREIBUNG II
DESCRIPTION DU PROGRAMME II
DESCRIZIONE DEL PROGRAMMA II

Example
Beispiel
Exemple
Esempio

YOU PLAY WITH 5 POSITIONS AND 9 COLOURS (1-9).

HP SHOULD FIND YOUR COMBINATION : 72618

(IF YOU LIKE TO TRANSFER THE GAME TO A MASTERMIND-BOARD, USE THE FOLLOWING DEFINITION :

DIGIT	≙	COLOUR
1		FREE PLACE
2		WHITE
3		YELLOW
4		ORANGE
5		RED
6		GREEN
7		BLUE
8		BROWN
9		BLACK

)

Sketch
Skizze
Croquis
Schema

(Example continued)

PROGRAM DESCRIPTION III

PROGRAMMBESCHREIBUNG III
DESCRIPTION DU PROGRAMME III
DESCRIZIONE DEL PROGRAMMA III

	Data Input	Keystrokes	Result	Comments
		YEQ "ISUP"	SIZE P.C ?	P : NUMBER OF POSITIONS
				C : " " COLOURS
	5.9	R/S	SN 0<X<1	GIVE IN A Start-Number X
				WITH 0<X<1
	.284117	R/S	77777 + 1	1. GUESS
			GUESS No	
INFO :	1.0	R/S	44447 + 2	2. "
"	0.1	R/S	11171 + 3	3. "
"	0.2	R/S	99719 + 4	4. "
"	1.1	R/S	67616 + 5	5. "
"	2.1	R/S	75516 + 6	6. "
"	2.1	R/S	73613 + 7	7. "
"	3.0	R/S	78618 + 8	8. "
"	4.0	R/S	72618 + 9	9. "
"	5.0	R/S	72618 + 9	SOLVED

USER INSTRUCTIONS I

PROGRAMMABLAUF I
INSTRUCTIONS D'EMPLOI I
NORME OPERATIVE I

Step	Instructions	Variables	Function(s)	Result
1	INPUT PROGRAM			
2	START		XEQ "ISUP"[1]	SIZE P.C :
3	INPUT NUMBER P OF POSITIONS (P ≤ 9) AND NUMBER C OF COLOURS (C ≤ 9) YOU HAVE CHOSEN TO PLAY WITH	P.C	R/S	SN $0 < X < 1$
4	INPUT AN OPTIONAL STARTNUMBER X BETWEEN 0 AND 1	X	R/S	1 GUESS (FORM: GUESS * No.)
5	INPUT YOUR INFORMATION IN FORM OF BLACK (B) AND WHITE (W) KEY PEGS	B.W	R/S	NEXT GUESS
	REPEAT STEP (5)			
	IF THE CALCULATOR HAS DECODED YOUR COMBINATION COMPLETELY (INFO: P.0) HE WILL SHOW THE LAST GUESS AND THE NUMBER OF GUESSES AGAIN.			
6	FOR A NEW GAME GO TO (2)			
	[1] YOU CAN ASSIGN "ISUP" TO ANY KEY			

PROGRAM LISTING

PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

Line	Keystrokes Tastenfolge Touches Tasti	Comments Kommentar Commentaires Commenti	Line Zeile Ligne Linea	Key pressed Tastenfolge Touches Tasti	Comments Kommentar Commentaires Commenti
01	* LBL "ISUP"		051	ST + 26	
	CF 29			+	
	FIX 0			STO 27	
	CLRG			* LBL 02	
05	.008		55	CLX	
	STO 00			STO 28	
	* LBL 00			* LBL 03	
	SF IND 00			RCL IND 00	
	ISG 00			INT	
10	GTO 00		60	X ≠ 0 ?	
	"_ _ SIZE _ P.C _ ?"			GTO 04	
	PROMPT			RCL 27	
	INT	Number of		STO IND 00	
	STO 20	Positions		* LBL 04	
15	LAST X		65	RCL 28	COMPOSITION
	FRC			10 ↑ X	OF A GUESS
	10			x	
	x			ST+ IND 33	
	STO 23	Number of		1	
20	.1	Colours	70	ST+ 28	
	STO 24			DSE 00	
	11	used for		GTO 03	
	STO 25	indirect		ST+ 10	
	33	operations		* LBL 05	
5	STO 33		75	CLA	
	"_ _ SN _ 0<X<1"			ARCL IND 33	
	PROMPT			⊢ X	
	STO 21	Start-Number		ARCL 10	
	* LBL 17			AVIEW	DISPLAY
0	1		80	STOP	
	ST + 33			RCL 20	
	RCL 20			STO 00	SECRET CODE
	STO 00			X = Y ?	COMPLETELY
	RCL 23			GTO 05	DECODED ?
5	RCL 26	ALL COLOURS	85	R↓	
	X = Y ?	GENERATED ?		10	
	GTO 02			/	
	* LBL 01			ST+ IND 33	Routine
	RCL 21	RANDOM-DIGIT		10	works up
0	PI	GENERATION	90	x	Information
	+			ENTER ↑	
	E↑X			FRC	
	FRC			10	
	STO 21	NEW 0<X<1		x	
5	RCL 23		95	+	
	x			INT	
	INT			STO 32	
	FC?C IND X	ONLY DIFFE-		ENTER ↑	
	GTO 01	RENT COLOURS		X<>22	
	1		100	-	

PROGRAM LISTING
PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

Line Zeile Ligne Linea	Keystrokes Tastenfolge Touches Tasti	Comments Kommentar Commentaires Commento	Line Zeile Ligne Linea	Key pressed Tastenfolge Touches Tasti	Comments Kommentar Commentaires Commento
1 01	STO 27		1 51	STO 31	
	X=0 ?	NEW COLOURS ?		ST+ 28	
	GTO 08			X=0 ?	
	♦ LBL 06			GTO 20	
05	RCL IND 00		55	0	
	FRC			STO 29	
	X=0 ?			RCL 20	
	GTO 07			STO 00	
	1			♦ LBL 09	
10	ST- 00		60	RCL 34	
	GTO 06			INT	
	♦ LBL 07			10	
	RCL 24			/	
	ST+ IND 00			STO 34	
15	RCL IND 00		65	FRC	
	STO IND 25			10	
	1			x	
	ST- 00			RCL IND 00	
	ST+ 25			INT	
20	.1		70	X≠Y ?	
	ST+ 24			GTO 10	
	DSE 27			1	
	GTO 06			ST+ 29	
	♦ LBL 08			ST- 30	
25	RCL IND 00		75	RCL 30	
	FRC			X<0 ?	
	X=0 ?			GTO 21	
	STO IND 00			♦ LBL 10	
	DSE 00			DSE 00	
30	GTO 08		80	GTO 09	
	♦ LBL 18			RCL 32	
	RCL 33	IS THERE		RCL 28	
	.034	A CONTRADIC-		X>Y ?	
	+	TION ?		R↓	
35	STO 27		85	RCL 29	
	♦ LBL 19			-	
	RCL IND 27			CHS	
	ENTER ↑			RCL 31	
	INT			+	
40	STO 34		90	X<0 ?	
	-			GTO 21	
	10			DSE 27	
	x			GTO 19	
	ENTER ↑			♦ LBL 20	
45	INT		95	RCL 22	ARE ALL
	STO 30			RCL 32	GUESSES +
	STO 28			X=Y ?	COLOURS
	-			GTO 17	EXAMINED ?
	10			11	
1 50	x		2 00	+	

PROGRAM LISTING

PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

Line Zeile Ligne Linea	Keystrokes Tastenfolge Touches Tasti	Comments Kommentar Commentaires Commenti	Line Zeile Ligne Linea	Key pressed Tastenfolge Touches Tasti	Comments Kommentar Commentaires Commenti
2 01	STO 31		51		
	1				
	ST+ 32				
	RCL 20				
05	STO 00		55		
	* LBL 11				
	RCL IND 00				
	X=0 ?				
	GTO 12				
10	DSE 00		60		
	GTO 11				
	* LBL 12				
	RCL IND 31				
	STO IND 00				
15	GTO 18		65		
	* LBL 21				
	RCL 20	PERMUTATION			
	STO 00				
	* LBL 13				
20	RCL 32		70		
	10				
	/				
	RCL IND 00				
	FRC				
25	X=Y ?		75		
	GTO 14				
	DSE 00				
	GTO 13				
	* LBL 14				
30	0		80		
	X < > IND 00				
	STO 28				
	* LBL 15				
	DSE 00				
35	GTO 16		85		
	DSE 32				
	GTO 21				
	* LBL 16				
	RCL IND 00				
40	X≠0 ?		90		
	GTO 15				
	RCL 28				
	STO IND 00				
	GTO 18				
2 45	END		95		
50			100		

REGISTERS, STATUS, FLAGS

REGISTERBELEGUNG, FLAGS, BETRIEBSARTEN
REGISTRES, INDICATEURS, MODES OPÉRATOIRES
REGISTRI, MODI OPERATIVI, FLAGS

Registers / Datenspeicher / Registres de données / Registri

Reg.	Content	Reg.	Content
00	COUNTER	50	
01			
	DIGITS		
05	OF A	55	
	GUESS		
	(max. 9)		
09			
10	NUMBER OF GUESSES	60	
11			
15	marked	65	
	DIGITS		
	(max. 9)		
19			
20	COUNTER (Pos.	70	
21	0<X<1 RANDOM		
22	COUNTER		
23	" (Colours)		
24	COUNTER		
25	" (+	75	
26	"		
27	" other		
28	" functions)		
29	"		
30	"	80	
31	"		
32	"		
33	"		
34	USED		
35		85	
:			
:			
:	GUESSES		
:	+		
40	INFORMATION	90	
:			
:			
:			
:			
45		95	
:			
47			
		99	

Status / Betriebsart / Modes opératoires / Modi operativi

Size 048 Total Reg. 110 (62 + 48)

			User Mode
Eng ☐	Fix ☒ 0	Sci ☐	On ☐
Deg ☐	Rad ☐	Grad ☐	Off ☒

Purpose / Bedeutung / Signification / Scopo — **Flags**

Flag	Purpose	SET	CLEAR
00			
01			
02	USED		
03			
04	FOR GENERA-		
05	TION OF		
06	DIFFERENT		
07	COLOURS		
08			
09			
10			
11	Audio enable		
12			
13			
14			
15			
16			
17			
18			
19			
20			
21	Printer Enable		
22	Number Input		
23	Alpha Input		
24	Range Ignore		
25	Error Ignore		
26	Audio Enable		
27	User Mode		
28	Decimal Point		
29	Digit Grouping		X

Assignments / Tastenbelegung / Assignations / Assegnazioni

Function / Funktion / Fonction / Funzione	Key / Taste / Touche / Tasto	Function / Funktion / Fonction / Funzione	Key / Taste / Touche / Tasto

Superhirn für den HP-41C

Ein Spielprogramm für den programmierbaren Rechner HP-41C

Autor : Ulrich Davertzhofen , 1979/80

Das Spiel

Spieler A steckt x Codestecker in die x Löcher , ohne daß der Entschlüsseler B die geheime Kombination einsehen kann. Bei der Zusammenstellung des Geheimcode ist es erlaubt, auch 2 oder mehr Stecker der gleichen Farbe zu verwenden (z.B. rot, blau,rot,grün). Spieler B versucht nun, den Code exakt nachzubilden. Nach jedem Versucht gibt A Auskunft über den Stand der Dechiffrierung durch schwarze u. weiße Chiffrierstifte.
Dabei bedeutet :

1 schwarzer Stift : ein Codestecker entspricht nach Position und Farbe einem Stecker des Geheimcode ;
1 weißer Stift : ein Codestecker entspricht nach der Farbe, aber nicht nach der Position einem Stecker der geheimen Kombination .

Steckt A für alle x Positionen als Information schwarze Stifte, so ist der Code vollständig entschlüsselt und das Spiel beendet. Ziel ist es, den Code in möglichst wenig Rateversuchen zu finden.

Das Programm

Der Rechner übernimmt den Part von Spieler A , indem er (pseudo-) zufällig eine Ziffernkombination erzeugt, die der "menschliche Gegenspieler" B entschlüsseln muß. Der Geheimcode kann aus max. 9 Positionen (x) und max. 9 Farben (y) bestehen, wobei jeder Farbe eine Ziffer zugeordnet wird (etwa: 1= freie Stelle, 2= weiß , ... , 9= schwarz). Zu den einzelnen Rateversuchen gibt der Rechner die entsprechenden Informationen in der Form s.w (Anzahl schwarze.Anzahl weiße Chiffrierstifte) aus.

HP 41C PROGRAM SUBMITTAL FORM

PROGRAMMFORMBLATT/DOCUMENTATION DU PROGRAMME/GENERALITÀ SUL PROGRAMMA

Program Title (Programmtitel / Titre du programme / Titolo del programma): PLAYING MASTERMIND WITH HP-41 C

Category No. (Kategorie Nr / Catégorie N / Categoria N): 821 **Name** (Rubrik / Rubrique / Nome della categoria): BOARD- / TABLE GAMES

No. of program lines (Anzahl Programmzeilen / Nombre de lignes de programme / Nº di linee di programma): 108

No. of data registers (Anzahl der benötigten Datenspeicher / Nombre de registres de données / Nº di registri utilizzati): 41 (prg: 24, data: 17)

Recommended HP 41C System configuration (Empfohlene System Konfiguration / Configuration recommandée / Configurazione raccomandata)
Port # 1 ____ Port # 2 ____
Port # 3 ____ Port # 4 ____

This program requires the following programs as subroutines: (Dieses Programm benutzt folgende Programme als Unterprogramme / Ce programme utilise les programmes suivants comme sous-programmes / Questo programma usa i seguenti programmi come subroutines)

HP Applications ROM (HP Applikations ROM / ROM d'application HP / ROM di applicazione HP) ____ **Program Name:** (Programm / Nom du programme / Programma) ____

Program Abstract (Kurzbeschreibung / Résumé / Breve descrizione del programma):
YOUR CALCULATOR GENERATES A SECRET CODE WITH UP TO 9 POSITIONS AND 9 COLOURS (1 - max. 9), AND IT IS YOUR TASK TO FIND IT OUT. AFTER EACH GUESS HP-41 TELLS YOU HOW MANY DIGITS ARE RIGHT IN POSITION AND COLOUR (BLACK KEY-PEG) AND HOW MANY ARE RIGHT IN COLOUR ONLY (WHITE). AT THE END YOUR HP SHOWS THE SECRET CODE AND THE NUMBER OF GUESSES YOU NEEDED FOR DECODING IT.

Name (Name / Nom / Nome): ULRICH DAVERTZHOFEN

Address (Strasse / Adresse / Indirizzo): HUENEFELDSTRASSE 109

City (Ort / Localité / Città): WUPPERTAL **Postal Code** (Postleitzahl / Code postal / C.A.P.): 5600 **Country** (Land / Pays / Paese): W-Germany

ACKNOWLEDGMENT AND AGREEMENT

Erklärung und Ermächtigung/Déclaration et Autorisation, Dichiarazione e Autorizzazione

To the best of my knowledge, I have the right to contribute this program material without breaching any obligation concerning nondisclosure or confidential information of other persons or organizations. I am contributing this program material on a nonconfidential nonobligatory basis to Hewlett-Packard S.A. ("HP") for inclusion in its program library, and I agree that HP may use, duplicate, modify, publish, and sell the program material, and authorize others to do so without obligation or liability of any kind. HP may publish my name and address, as the contributor, to facilitate user inquiries pertaining to this program material.

Ich versichere nach bestem Wissen, dass ich über meinen Programmbeitrag frei verfügen kann, ohne dass sich dadurch für HP, andere Programmbenutzer oder mich irgendwelche Verpflichtungen gegenüber Dritten oder sonstige rechtliche Nachteile ergeben. HP kann meinen Programmbeitrag ohne Geheimhaltungs- und sonstige Verpflichtungen in beliebiger Weise benutzen oder verwerten. Gegen eine Veröffentlichung meines Namens im Zusammenhang mit meinem Programmbeitrag habe ich nichts einzuwenden.

[illegible]

[illegible]

Date (Datum / Date / Data): Juni 1980 **Signature** (Unterschrift / Signature / Firma): Ulrich Davertzhofen

PROGRAM DESCRIPTION I

PROGRAMMBESCHREIBUNG I
DESCRIPTION DU PROGRAMME I
DESCRIZIONE DEL PROGRAMMA I

Application, Equations, Variables
Anwendung, Gleichungen, Veränderliche
Application, Equations, Variables
Applicazione, Equazioni, Variabili

YOU WANT TO DECODE A COMBINATION WITH
5 POSITIONS AND UP TO 9 COLOURS :

Operating limits and Warnings
Grenzen und Einschränkungen
Limites et restrictions
Limiti operativi e avvertenze

1) $P, C \le 9$

2) LOOK FOR CORRECT INPUT OF YOUR GUESS (ES) !

3) DO NOT PRESS RCL 01,..., RCL 09 !

PROGRAM DESCRIPTION II

PROGRAMMBESCHREIBUNG II
DESCRIPTION DU PROGRAMME II
DESCRIZIONE DEL PROGRAMMA II

Example / Beispiel / Exemple / Esempio	DATA INPUT	KEYSTROKES	RESULT	COMMENTS
	5.9	ENTER		P: NUMBER OF POSITI
	.721835	XEQ "SUP"	SIZE P.C = 5.9	C: " " COLOU
			0.0	CODE IS GENERATED
1. GUESS	12345	R/S	1.2	1 BLACK, 2 WHITE KEY PEG
2. "	67899	R/S	0.2	
3. "	13267	R/S	0.3	
4. "	11111	R/S	0.0	
5. "	33333	R/S	1.0	
6. "	22222	R/S	0.0	
7. "	35746	R/S	35746 . 7	"BEEP"
			7 guesses	SOLVED

Sketch
Skizze
Croquis
Schizzo

USER INSTRUCTIONS I

PROGRAMMABLAUF I
INSTRUCTIONS D'EMPLOI I
NORME OPERATIVE I

Step	Instructions	Variables	Function(s)	Result
1	Input program			
2	Input number P of positions (P≤ 9) and number C of colours (C≤ 9)	P.C	Enter ↑	
3	Input an optional start-number between 0 and 1	x	XEQ 'SUP' *)	SIZE P.C = chosen Size 0.0 +)
4	Input your guess	guess	R/S	information b.w ++)
5	Repeat (4)			
	When you have decoded the secret combination completely HP shows the code (= your last guess) and the number of guesses for decoding it.			code x no. Beep
6	for a new game go to (2)			

+) 0.0 shows that the secret code is generated.

++) b.w : information in form of black(b) and white(w) key-pegs

*) YOU CAN ASSIGN 'SUP' TO ANY KEY

PROGRAM LISTING

PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

Line / Zeile / Ligne / Linea	Keystrokes / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commento	Line / Zeile / Ligne / Linea	Key pressed / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commento
0 01	• Lbl "SUP"		0 51	X ≠ Y ?	
	CLRG			GTO 03	
	CF 29			1	
	FIX 1			ST+ 11	
05	STO 01		55	CHS	
	R↓			ST× IND 00	
	"SIZE_P.C_=_"			• Lbl 03	
	ARCL X			DSE 00	
	AVIEW			GTO 02	
10	INT		60	RCL 12	
	STO 00	} P		STO 13	
	STO 14			RCL 14	
	LAST X			STO 00	
	FRC			• Lbl 04	
15	10		65	RCL 14	TEST FOR
	×			STO 10	WHITE KEY-
	STO 12	C		XEQ 09	PEGS
	• Lbl 00			RCL IND 00	
	RCL 01	GENERATES		X < 0 ?	
20	PI	SECRET CODE	70	GTO 07	
	+			• Lbl 05	
	e^x			RCL 15	
	FRC			RCL IND 10	
	STO 01			X ≠ Y ?	
25	RCL 12		75	GTO 06	
	×	DIGITS		.1	
	1	> 1		ST+ 11	
	+			ST+ IND 10	
	INT			GTO 07	
30	STO IND 00		80	• Lbl 06	
	DSE 00	COUNTER		DSE 10	
	GTO 00			GTO 05	
	• Lbl 01			• Lbl 07	
	RCL 14			DSE 00	
35	X < > 00		85	GTO 04	
	X < > 11			1	
	RCL 14			ST+ 16	
	X = Y ?			GTO 01	GOTO DISPLAY
	GTO 08			• Lbl 08	
40	R↓		90	FIX 0	
	CLD			CLA	
	STOP	DISPLAY INFO		ARCL 12	CODE
	STO 12	}		⊢x	
	STO 13	} GUESS		ARCL 16	NUMBER OF GUES-
45	• Lbl 02		95	AVIEW	SES
	XEQ 09			BEEP	
	RCL IND 00	TEST FOR		STOP	END
	ABS	BLACK KEY-		• Lbl 09	SUBROUTINE
	INT	PEGS		RCL 13	
0 50	STO IND 00		1 00	INT	

PROGRAM LISTING
PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

Line / Zeile / Ligne / Linea	Keystrokes / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commento	Line / Zeile / Ligne / Linea	Key pressed / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commento
1 01	10	SEPERATES	51		
	/	DIGITS FOR			
	STO 13	TESTS			
	FRC				
05	10		55		
	x				
	STO 15				
108	END				
10			60		
15			65		
20			70		
25			75		
30			80		
35			85		
40			90		
45			95		
50			100		

REGISTERS, [illegible]
REGISTERBELEGUNG, FLAGS, BETRIEBSARTEN
REGISTRES, INDICATEURS, MODES OPÉRATOIRES
REGISTRI, MODI OPERATIVI, FLAGS

Registers
Datenspeicher
Registres de données
Registri

Reg.	Content	Reg.	Content
00	COUNTER	50	
01	}		
	DIGITS		
	OF		
05	SECRET	55	
	CODE		
	(MAX. 9)		
09	}		
10	COUNTER	60	
11	INFORMATION		
12	} GUESS		
13	}		
14	COUNTER (P)		
	USED	65	
16	NUMBER OF		
	GUESSES		
20		70	
25		75	
30		80	
35		85	
40		90	
45		95	
		99	

Status
Betriebsart
Modes opératoires
Modi operativi

Size 017 Total Reg. 41 (24 + 17)

						User Mode	
Eng	☐	Fix	☒ 0/1	Sci	☐	On	☐
Deg	☐	Rad	☐	Grad	☐	Off	☒

Flags	Purpose Bedeutung Signification Scopo	SET	CLEAR
00			
01			
02			
03			
04			
05			
06			
07			
08			
09			
10			
11	Auto execute		
12			
13			
14			
15			
16			
17			
18			
19			
20			
21	Printer Enable		
22	Number Input		
23	Alpha Input		
24	Range Ignore		
25	Error Ignore		
26	Audio Enable		
27	User Mode		
28	Decimal Point		
29	Digit Grouping		X

Assignments
Tastenbelegung, Assignations, Assegnamenti

Function Funktion Fonction Funzione	Key Taste Touche Tasto	Function Funktion Fonction Funzione	Key Taste Touche Tasto

Edmund Weitz
Heinr.-Heine-Str. 101

3151 Essinghausen

Essinghausen, 22.07.80

Herrn
Oliver Rietschel
Postfach 373

2420 Eutin

Hallo Oliver,

hier habe ich einen kleinen Vorschlag zur Programmänderung für das Games Solution Book. Er betrifft das Programm Wari. Hier habe ich zwei Punkte zu "bemängeln":

- Egal, wie der Seed eingegeben wird, er wird durch Clrg immer wieder auf 0 gesetzt

- Ein Fehler in den Spielregeln: Es ist nicht erlaubt, aus einer leeren Kuhle Steine zu "verteilen", d.h. eigentlich gar keinen Zug zu machen, obwohl das alte Programm dies zuläßt.
  Das neue Programm (also nach der Änderung) entscheidet hingegen, ob der Zug illegal war (wenn noch Löcher besetzt sind) oder ob das Spiel beendet werden muß (wenn alle Löcher der ausführenden Seite leer sind).

Folgende Änderungen müssen gemacht werden:

Positionieren des Rechners auf Zeile 08 (Lbl E) mit Gto.008.
Einfügen der folgenden Schritte: Rcl 00 Clrg Sto 00
Zweimal Sst ausführen
Löschen von Schritt 13 (Clrg) durch ←
Einfügen von Enter↑
Prgm-Modus aus
Gto B
Prgm-Modus wieder ein
Dreimal Sst ausführen
Einfügen: Sf 10 und Gto 12
Sst
Einfügen: Cf 10 und Lbl 12
Gto.175 (Gto 50)
Löschen von Gto 50 durch ←
Einfügen von Gto D
Zum Ende des Speichers gehen mit Gto.999
Einfügen von Rtn und folgenden Programmteil hinter Rtn einfügen:

```
Lbl D    Fc? 10    1.006    Fs? 10    7.012    "ILLEGAL"    Aview    Lbl 10
Rcl Ind X    X≠0?    Gto 11    Rdn    Isg X    Gto 10    "ENDE"    Cf 26
Lbl 11    Beep    Sf 26    Prompt
```

Fertig!! Das neue Programm benötigt zur Aufzeichnung auf Karte einen Track mehr als das alte, also 5 Tracks.

Happy Programming

Formel 1 für HP-41C

Kurzbeschreibung:

Der Spieler hat einen Formel-1-Wagen und sieht eine Rennstrecke vor sich. Im Gegensatz zum Rennfahrerspiel muß er auch noch lenken. Das Spiel ist sicherlich eins der interessantesten, daß je für einen Taschenrechner geschrieben wurde.

Referenz:

HP-67-Programm von Jean Thiberge
Ciram
50107 Cherbourg
Frankreich

Beschreibung der Anzeige:

G . l l l l o r r r r

G= Gang
l= linker Streckenteil
o= Fahrzeugnase
r= rechter Streckenteil

Beispiel:

1.111501111

Rechtskurve mit Limit (5) in 3 sec (Jede Kurve wird 3 sec in voraus angezeigt)

1.155501111

Beginn dieser Rechtskurve

3.411106661

Ende einer Rechts-, Beginn einer Linkskurve

Kurvenlimits (bedeutung der Kurvenzahlen)

Je höher die Zahl, je langsamer die Kurve

1	300
2	270
3	240
	...
9	30

Ganglimits

Gang	Limit
1	90
2	140
3	195
4	252
5	300

Speicherbelegung:

0=	seed	1=	nächste Kurve	2=	aktuelle Kurve
3=	Beschleunigung	4=	linke Seite	5=	Mitte
6=	rechte Seite	7=	Strecke	8=	Tempo
9=	Zeit	10=	Tempolimit (Gang)		
11=	Gang	12=	gerade Strecke	13=	Länge der Strecke
14=	Beschleunigung in km/h/sec			15=	benutzt

Bedieunungsanleitung

Schritt	Anweisung	Input	Tasten	Output
1	Programm laden			
2			XEQ "F1"	"DISTNAZ?"
3	Distanz eingeben	D m	R/S	"SEED?
4	Seed eingeben	S	R/S	
	--→ Countdown			-3.111101111 -2.111101111 -1.111101111 0.111101111
	1. Gang einlegen		1	
5	jede Sekunde:			Rennstrecke Tempo
6	Während dieser Anzeige kann man			
a)	gleiches Tempo		C	
b)	Vollbremsung		b	
c)	1/3 Bremsung		B	
d)	Vollgas		d	
e)	1/3 Beschleunigung		D	
f)	Linkskurve		A	
g)	Rechtskurve		E	
Nur als letzte dieser Aktionen:				
h)	Gang einlegen	Gang		
7	Rennende			Zeit Distanz Tempo bei Ziel Durchschnitts-tempo
8	Fehlermeldungen			
	Zu schnell für jeweiligen Gang			Überdreht
	Zu langsam			Abgewürgt
	Keine 1 am Start getippt			Start verschla-fen
	In Kurven zu schnell oder falsch gelenkt			Unfall
9	Um die Anzeigen (End anzeigen) noch einmal zu betrachten:		R/S	
10	Für Neustart mit gleicher Distanz: J			



```
01 LBL "F1"        Start des Programms
   "DISTANZ?"      Länge der Rennstrecke in Metern
   PROMPT
   STO 13
   "SEED?"         Zufallszahl (zur Erzeugung einer zufälligen
   PROMPT          Kurvenfolge)
   GTO c           Zum Countdown
08 LBL 00          Aufbau der Anzeige
   10
   ST/ 06          Rechte Seite eine Position nach rechts schieben
   RCL 04
   x               Linke Seite eine Position nach links schieben
   FRC             Verhindern, daß linke Seite in den Gang hinein-
   RCL 05          geschoben wird
   +               Mitte aktualisieren
   RCL 06
   +               Gesamtstreckenabschnitt
   RCL 11
   +               Gang addieren, Anzeige fertig
   ENTER^
   FRC
   STO 06
   FIX 4
   RND             Hintere sechs Stellen abschneiden
   STO 04          Neuer linker Abschnitt
   ST- 06          Neuer rechter Abschnitt
   CLX
   RCL 12
   +               Gerade (111101111) addieren
   FIX 9           Anzeigeformat
31 LBL 10
   PSE
   PSE             2 sec anzeigen
   FIX 1           Anzeigeformat für Tempo
   CLA
   ARCL 08
   "⊢ KM/H"
   AVIEW
   PSE
   PSE             Tempo zwei sec anzeigen
   CLD             Anzeige löschen
42 LBL 01
   FS?C 22         Lag eine Eingabe (Gangschalten) vor?
   XEQ 02          Wenn ja, Gang einlegen
   1
   ST+ 09          1 sec addieren
   RCL 08
   ST+ 07          Gefahrene Strecke addieren
   RCL 10
   /
   X>Y?            Zu schnell (Motor überdreht?)
   GTO 09
   RCL 14
   X>0?
   P-R             Effektive Beschleunigung, falls Beschleunigung
   ST+ 08          positiv ist
   RCL 08          neues Tempo
   ST+ 07          Zur gefahrenen Strecke addieren
   SF 25           Fehlermeldung verhindern
   SQRT
   FC?C 25         Tempowert negativ?
```

```
    GTO 08          Fehlerroutine
    10
    RCL 02          Kurvencode
    FS? 00
    CHS
    X<0?            Korrekt gelenkt?
    GTO 04          Wenn nicht, Unfall
    -
    30
    x               Tempolimit für Kurve errechnen
    RCL 08
    X>Y?            Zu schnell?
    GTO 04          Wenn ja, Unfall
    DSE 15          Ist eine neue Kurve nötig (nach 3 sec)
    GTO 00          Wenn nicht, Kurvenerzeugung überspringen
    3
    STO 15          Kurvenzeitzähler wieder auf 3 sec stellen
    RCL 01
    STO 02          Zuletzt gefahrene Kurve in den zur Kontrolle be-
    1               nutzten Speicher abspeichern
    RCL 00          Zufallszahlengenrator
    9821
    x
    .211327
    +
    FRC
    STO 00          Neue Zufallszahl abspeichern
    17
    x
    INT
    8
    -               Erzeugung ganzer Zahlen zwischen +9 und -9
    STO 01          Als nächste Kurve abspeichern
    X<0?            Wenn negativer Wert, Linkskurve
    %               In diesem Falle um 2 Stellen nach rechts schieben
    ABS             Wert ohne Vorzeichen
    4 E-6
    -
    1 E4
    /
    STO 05          Als mittleren Streckenteil speichern
103 LBL 00
    RCL 07
    7.2             Strecke von km in n Stunden in m in n sec um-
    /               rechnen
    RCL 13
    X>Y?
    GTO 00
    2               Wenn gefahrene Strecke vorgegebene Distanz über-
    ST/ 07          schritten hat, Endanzeige
    RCL 09
    ST/ 07
    R↑
    RCL 08
    RCL 07
117 LBL 11
    CLA
    ARCL T
    "⊦ SEC"
    PROMPT
    CLA
    ARCL Z
```

⑤

```
     SF 22
188  LBL D           1/3 Beschleunigung
     288
190  LBL 03          Errechnen der neuen (alten) Beschleunigung
     3
     FS?C 22         Mit 3 oder mit 9 malnehmen
     X↑2
     x
     STO 03
     STO 14
     X>0?            Positive Beschleunigung,
     XEQ 03          Dann abhängig vom Gang
     RDN
     FS?C 05
     RTN             Wenn das Upro aufgerufen wurde, Rücksprung
     GTO 10
203  LBL A           Nach links lenken
     SF 00
     PSE
     GTO 10
207  LBL E           Nach rechts lenken
     CF 00
     PSE                   ↓
     GTO 10
211  LBL 02             248   BEEP
     ENTER↑                   RTN
     STO 11                   .END.
     1
     +
     1/X
     +
     60
     x               Limit für jeweiligen Gang berechenen
     STO 10
     RCL 03
     X<0?
     RTN
224  LBL 03          Evtl. neue effektive Beschleunigung berechnen
     RCL 10
     /
     STO 14
     RTN
229  LBL 04
     "UNFALL"
     AVIEW
     BEEP
     RTN
234  LBL 09
     "UEBERDREHT"
     AVIEW
     BEEP
     RTN
239  LBL 08
     "ABGEWUERGT"
     AVIEW
     BEEP
     RTN
244  LBL 07
     "START VERSCHLAF"
     "├EN"
     AVIEW
```

```
    "⊦ M"
    PROMPT
    CLA
    ARCL Y
    "⊦ KM/H"
    PROMPT
    "SCHNITT="
    ARCL X
    "⊦ KM/H"
    PROMPT
    GTO 11
135 LBL J
    RCL 00
137 LBL c
    RAD
    SF 05
    SF 27             USER-Mode an
    CF 28
    CF 29
    RCL 13
    CLRG
    STO 13
    RDN
    STO 00
    1
    STO 10
    FIX 9
    3
    STO 15
    9
    1/X
    1 E-5
    -                 Erzeugen der Geraden .111101111
    STO 12            Abspeichern
    +
    CHS               Countdown vorbereiten
    XEQ d
    +
    PSE
    +
    PSE
    RCL 12
    PSE               Ende des Countdowns
    SF 25
    INT
    1/X
    FC?C 25           Eingegebener Wert nicht 1?
    GTO 07
    SF 25
    INT
    1/X
    FC?C 25           dto.
    GTO 07
    GTO 01            Zur Anzeigeroutine
178 LBL b             Vollbremsung
    SF 22
180 LBL B             1/3-Bremsung
    -6
    GTO 03
183 LBL C             Konstantes  Tempo
    0
    GTO 03
186 LBL d             Vollgas!!
```

⑥

Programmname:

Biorhythmus - Ergänzung zu 'Kalenderrechnungen'

Entwicklung:

Oliver Rietschel
Bismarckstr. 31
2420 Eutin
W. Germany

09/02/80

Das folgende Programm stellt ein Unterprogramm zu'Kalenderrechnungen' aus der Standardprogrammsammlung dar. Folgende Eingabe:

- Geburtsdatum eingeben -- XEQ A
- Heutiges Datum " -- XEQ B
- Differenz in Tagen berechnen -- XEQ C
- Biorhythmus'berechnen -- XEQ D

```
LBL D          LBL 00
FIX 2          /
DEG            360
23             *
XEQ 00         SIN
KRAFT:         10
ARCL X         *
AVIEW          RTN
STOP
RCL 01
28
XEQ 00
EMOTION:
ARCL X
AVIEW
STOP
RCL 01
33
XEQ 00
GEIST:
ARCL X
AVIEW
FIX 0
RTN
```

Benötigter Speicherplatz : 10 Register

Programm: Entfernungsberechnung mit Koordinaten

a) Beschreibung

Dieses Entfernungsberechnungsprogramm ist ein speziel auf dem HP-41 C entwickeltes Anwenderprogramm, das nach der Formel

$$E_{(miles)} = \cos^{-1}\left[\sin(NL_1)\ \sin(NL_2) + \cos(NL_1)\ \cos(NL_2)\ \cos(OB_2 - OB_1)\right] \times 60$$

abläuft.
Die Entfernung ("E") ist eine Angabe in Miles. Für die umrechnung in km muss mit 1.8544 multipliziert werden.

Ich habe dieses Programm erstmals am 30. Juni 1980 geschrieben. Die jetzt vorliegende Fassung geht auf den 30. Juli 1980 zurück.

Noch einige Bemerkungen vorweg: Bei einem Input von negativen Werten wird richtig über den 0-Meridian hinweg gerechnet. Vorsicht: Bei Input von zweimal dem gleichen Punkt wird 90 m Entfernung berechnet. Dies ist jedoch falsch.

Das Programm verlangt nun bei Begin die Eingabe von 4 Positionen (genauer 2 mal x und y). Während der Verarbeitung gem. der oben erwähnten Formel zeigt der Rechner im Anzeigefeld die Bemerkung "Berechnung". Programmschritt 36 und 37 können bei einer Ausgabe in Miles weggelassen werden, doch empfehle ich dann den Pgrm.schritt 40 (" KM") durch (" MILES") zu ersetzen.

Wenn nun die Anzeige " 999.99 KM " im Display steht, kann das Programm mit (R/S) weitergeführt werden. Falls man jetzt aber mit diesem Wert rechnen will, genügt es, die Clear-Taste (←) zu bedienen. Die nächste Anzeige ist nun "NOCH EINMAL ?". Wenn dort "Y" ("Yes") eingegeben wird, verzweigt das Prgm. wieder zum Start; wenn nicht, dann kommt die Frage "INPUT-VIEW ?" (Anzeigen der Eingaben). Auch hier kann wieder mit "Y" geantwortet werden, dann zeigt die Anzeige für eine sec. "KONTROLLE : " und bringt dann für jeweils eine sec. die Werte (1-4) in dieser Form in den Display: "1. 99.99" / "2. 99.99" /"3. 99.99" / "4. 99.99". Anschliessend kommt die Frage "ZUM ANFANG ?" in die Anzeige. Wenn nicht "Y" eingegeben wird, zeigt der Rechner " * ENDE * ".

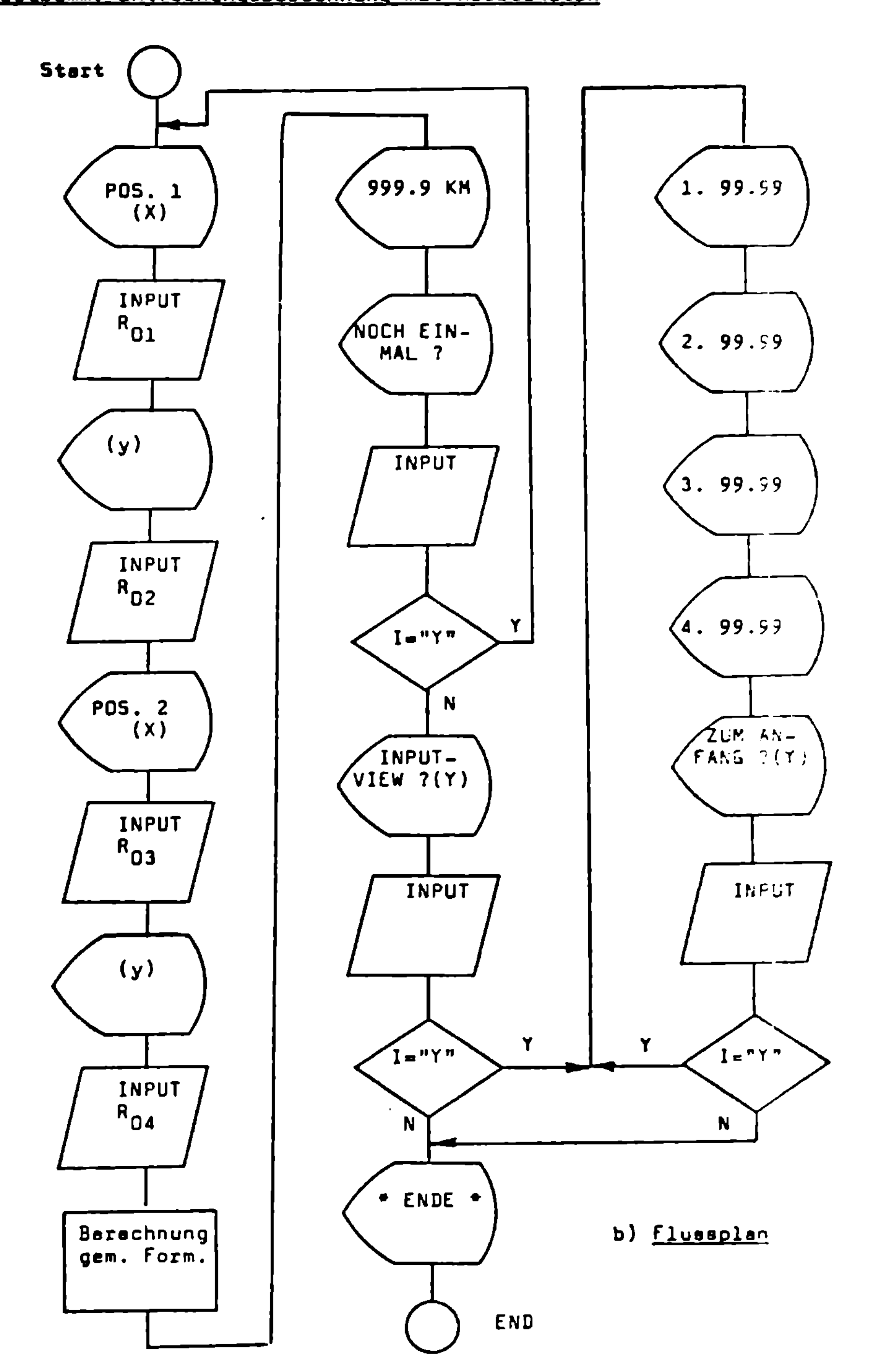
Das im LBL 07 verwendete Zeichen "Y" kann natürlich auch mit der deutschen Eingabe "J" ausgewechselt werden. Ich persönlich ziehe jedoch "Y" vor, weil es unter dem "*"-Zeichen sehr günstig liegt und einfach zu merken ist.

Auch der Name für das LBL 07 ist sehr willkürlich gewählt. Ich verwende in allen meinen Prgm. XEQ 07 für die Verarbeitung von "Yes/NO" Eingaben.

Das ganze Prgm. kommt auf 33 belegte Prgm.Zeilen, was 95 Programmschritten entspricht. Es werden die Speicher 00-05 und das Alpha-Reg. benützt.

Ueber Ideen, wie sich mein Programm verbessern oder verkürzen liesse, wäre ich sehr dankbar.

Programm: Entfernungsberechnung mit Koordinaten



b) Flussplan

Programm: Entfernungsberechnung mit Koordinaten

c) Listing

```
01 +LBL "ENTF
02 +LBL 01
03  AOFF
04  "POS. 1 (X)
05  PR.
06  STO 01
07  "(Y)
08  PR.
09  STO 02
10  "POS. 2 (X)
11  PR.
12  STO 03
13  "(Y)
14  PR.
15  STO 04
16  "BERECHNUNG
17  AVIEW
18  ENTER /
19  RCL 02
20  -
21  COS
22  RCL 03
23  COS
24  *
25  RCL 01
26  COS
27  *
28  RCL 01
29  SIN
30  RCL 03
31  SIN
32  *
33  +
34  ACOS
35  60
36  *
( 37  1.8544 )
38  *
39  STO 00
40  " KM
41  ASTO 05
42  CLA
43  ARCL 00
44  ARCL 05
45  PR.
46  "NOCH EINMAL?
47  AON
48  PR.
49  XEQ 07
50  X=Y?
51  GTO 01
52  "INPUT-VIEW?
53  PR.
54  XEQ 07
55  X≠Y?
56  GTO 03
57 +LBL 02
58  "KONTROLLE :
59  AVIEW
60  PSE
61  CLA
62  "1.‿
63  ARCL 01
64  AVIEW
65  PSE
66  CLA
67  "2.‿
68  ARCL 02
69  AVIEW
70  PSE
71  CLA
72  "3.‿
73  ARCL 03
74  AVIEW
75  PSE
76  CLA
77  "4.‿
78  ARCL 04
79  AVIEW
80  PSE
81  "ZUM ANFANG?
82  PR.
83  XEQ 07
84  X=Y?
85  GTO 01
86 +LBL 03
87  "‿**‿ENDE‿**‿
88  PR.
89 +LBL 07
90  ASTO Y
91  CLA
92  "Y
93  ASTO X
94  RTN
95  END
```

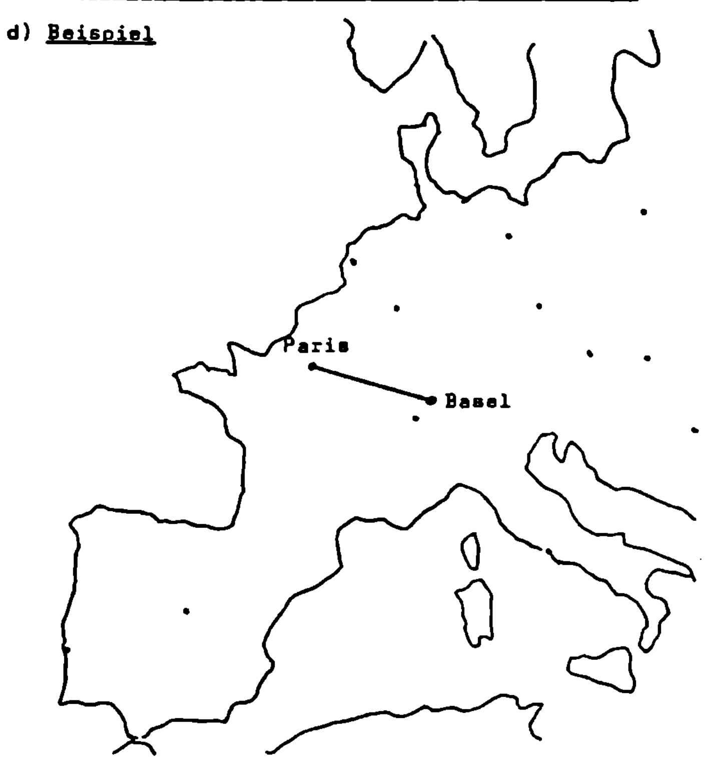
Erklärung:

(/) : Klammern (ich stelle diese mit dem "grösser als" und dem "kleiner als"-Zeichen dar - Shift 'I' ='(', Shift 'J' =')')

67 "2.‿ : Zeilennummer / '"' = normalerweise ein halbes "T"(hab ich aber nicht auf der Schreibmaschine) / 2. /'‿ ' = SPACE

Programm: Entfernungsberechnung mit Koordinaten

d) **Beispiel**



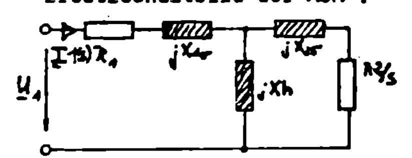
X = Nördliche Länge	Y = Oestliche Breite
-X = Südliche Länge	-Y = Westliche Breite

Paris : 48.833 / 2.333   Basel : 47.566 / 7.583

Eingabe:	Anzeige:	
XEQ "ENTF"	POS. 1 (X)	
48.833 (R/S)	(Y)	
2.333 (R/S)	POS. 2 (X)	
47.566 (R/S)	(Y)	
7.583 (R/S)	BERECHNUNG	(ca. 2 sec.)
	413.98 KM	
(R/S)	NOCH EINMAL?	
N (R/S)	INPUT-VIEW?	
Y (R/S)	KONTROLLE :	) jeweils ca. 1 sec.
	1. 48.83	)
	2. 2.33	)
	3. 47.57	)
	4. 7.58	)
	ZUM ANFANG ?	
N (R/S)	** ENDE **	

GRUNDLAGEN DES PROGRAMMS A S M

Ersatzschaltbild der ASM :



Definitionen :

$$S = \frac{n_s - n}{n_s} \quad ; \; n_s = f/PS$$

PS : Polpaarzahl

$$X1 = X1\sigma + XH$$
$$X2 = X2\sigma + XH$$

Alle Werte auf Ständerseite bez.

Komplexe Berechnung des Ständerstromes I1 :

$$I1(S) = \frac{U1}{R1 + jX1 + \frac{XH^2}{R2/S \; + \; jX2}}$$

Leistungsformeln :

PEL=3*U1*I1*COSF
$PV1 = 3 * R1 * I1^2$
PDEL=PEL-PV1
PMECH=PDEL(1-S)
PV2=PDEL*S

Wirkungsgrad :

ETA=PMECH/PEL

Drehmomentberechnung :

$$MeL(S) = \frac{3 * PSTR * R2^{+} * U1^2}{\omega * S\left((R1 + R2^{+}/S)^2 + X\emptyset^2\right)}$$

mit :

$$R2^{+} = UeK^2 * R2 = \frac{R1^2 + X1^2}{XH^2} * R2$$

$$X\emptyset = X2 * UeK^2 - X1 \; ; \; \omega = 2 * PI * f_{Netz}$$

Frequenzänderung :

$$X(f\;n+1) = \frac{(f\;n+1)}{50} * KOEFF. * X(f\;n)$$

mit :

KOEFF(0)=1
KOEFF(n+1)= 50/(f n)

!! f darf nicht Ø sein !!

ERLÄUTERUNGEN ZUM PROGRAMM A S M

Der maßgebende Programmteil ist die Berechnung des Ständerstromes I1.
Aus diesem werden alle Leistungen sowie COSF und ETA berechnet.
Die Ermittlung des Momentes MeL erfolgt direkt, könnte aber leicht auch aus PMECH und der Drehzahl erfolgen.
Der Schlupf zu einem gegebenen Strom I1 (Betrag) wird durch ein iterat. Verfahren ermittelt; ein vorbereitendes Programmstück nähert vorher die Kurve s=f(I1) durch eine e-Fkt. an. Die so gewonnenen Koeffizienten wirken konvergenzbeschleunigend.

PROGRAMMLAUF:

(Rechner fragt Daten der Masch. ab)

AUSGABE (seq. mit R/S)

NS - Synchrondrehzahl
a0 - Winkel um den Osannakreis gedreht ist
XØ - Durchmesserreaktanz
IØ - " strom
UeK - komplexes Übersetzungsverh.
SK - Kippschlupf
MK - " moment (Mot,Gen)
I1 - Ständerstom als Fkt(s)
COSF - Leistungsfaktor
PEL - Ständerleistung
PV1 - Ständerverlustleistung
PDEL - Luftspaltleistung
PMECH - mech. Leistung (abgeg.,zugef.)
ETA - Wirkungsgrad
PV2 - Läuferverlustleistung
MeL - entwickeltes Moment
FNEU - Routine zur Frequenzänderung!!
V - Ermittlung der Koeffizienten
S - Ermittlung von s bei geg. I1



MASCHINENBEISPIEL :

PROMPT	INPUT		OUTPUT	
U1=?	220	R/S		
R1=?	,65	"		
X1=?	35,8	"		
X2=?	35	"		
R2=?	,71	"		
XH=?	33,9	"		
PS=?	2	"		
F=?	50	"		
I1N=?	16	"	NS=1500 1/MIN	
		"	a0=1,040 GRD	
		"	XØ=3,246 OHM	
		"	IØ=67,773 A	
		"	UeK=1,056	
		"	SK=0,2393	
		"	MKM=116,70 NM	
		"	MKG=-173,72 NM	
	1 STO 06	"	I1=67,56 A	
		"	COSF =0,4040	
		"	PEL=18014,8	
		"	PV1=8899,3	
		"	PDEL=9115,6	
		"	PMECH=0,0	
		"	ETA=0,00000	
		"	PV2=9115,6	
		"	MeL=58,03 NM	
	XEQ V		I1=26,71 A	Werte für Näherungskurve
			I1=67,56 A	
GEN=SFØØ		"	SN=0,052478	Iteration NENNSCHLUPF
		"	N=1421,3 1/MIN	
		"	I1=16,00 A	NENNWERTE (MOTOR)
		"	COSF=0,8504	
		"	PEL=8980,8	
		"	PV1=499,3	
		"	PDEL=8481,5	
		"	PMECH=8036,4	
		"	ETA=0,89485	
		"	PV2=445,1	
		"	MeL=53,99 NM	

REGISTERBELEGUNG :

00	U1
01	R1
02	X1
03	X2
04	R2
05	XH
06	S
07	XØ
08	R2+
09	PS
10	F
11	NS
12	I1N
13	PEL
14	PDEL
15	I1
16	COSF
17	KOEFF
18	1.Kurvenpkt.
19	2. "
20	ln a, a Näherungskurve
21	b

ITERATIVES VERFAHREN FÜR S=f(I)

Annahme : $S=a\cdot e^{bI}$ $1=a\ e^{bIanl}$, $Sx=a\ e^{bIx}$

liefert : $\ln a = \dfrac{\ln Sx}{1-\dfrac{Ix}{Ianl}}$

$b = -1/Ianl * \ln a$

Iteration : nach SN

Forderung : $I-I(Si) \overset{!}{=} 0$

$S0 = a\ e^{bI}$

$Si+1=Si*(1-(I(Si)-I)b)$

bis:
$Si-Si+1 \leq EPS\ (1E-03)$

Benutzte SUBROUTINEN : CABS,CADD,CSUB,CMULT,CDIV (siehe Beilage)



③

```
01♦LBL "ASM"
02 FIX 2
03 "U1=?"
04 PROMPT
05 STO 00
06 "R1=?"
07 PROMPT
08 STO 01
09 "X1=?"
10 PROMPT
11 STO 02
12 "X2=?"
13 PROMPT
14 STO 03
15 "R2=?"
16 PROMPT
17 STO 04
18 "XH=?"
19 PROMPT
20 STO 05
21 "PS=?"
22 PROMPT
23 STO 09
24 1
25 STO 17
26 "F=?"
27 PROMPT
28 STO 10
29 50
30 X*Y?
31 XEQ "FNEU"
32 "I1N=?"
33 PROMPT
34 STO 12

35♦LBL "NS"
36 FIX 0
37 RCL 10
38 RCL 09
39 /
40 60
41 *
42 STO 11
43 "NS="
44 ARCL X
45 "⊢1/MIN"
46 AVIEW
47 RTN

48♦LBL "a0"
49 FIX 3
50 RCL 01
51 RCL 02
52 /
53 ATAN
54 "a0="
55 ARCL X
56 "⊢ GRD"
57 AVIEW
58 RTN

59♦LBL "X0"
60 FIX 2
61 XEQ "UeK"
62 X↑2
63 RCL 03
64 *
65 RCL 02
66 -
67 STO 07
68 "X0="
69 ARCL X
70 "⊢ OHM"
71 AVIEW
72 RTN

73♦LBL "I0"
74 RCL 00
75 RCL 07
76 /
77 "I0="
78 ARCL X
79 "⊢ A"
80 AVIEW
81 RTN

82♦LBL "UeK"
83 FIX 3
84 RCL 01
85 RCL 02
86 XEQ "CABS"
87 RCL 05
88 /
89 "UeK="
90 ARCL X
91 AVIEW
92 RTN
93 X↑2
94 RCL 04
95 *
96 STO 08

97♦LBL "SK"
98 FIX 4
99 RCL 07
100 RCL 01
101 XEQ "CABS"
102 RCL 08
103 X<>Y
104 /
105 STO 06
106 "SK="
107 ARCL X
108 AVIEW
109 RTN

110♦LBL "MK"
111 FIX 2
112 XEQ "MeL"
113 "MKM="
114 ARCL X
115 "⊢ NM"
116 AVIEW
117 STOP
118 -1
119 ST* 06
120 XEQ "MeL"
121 "MKG="
122 ARCL X
123 "⊢ NM"
124 AVIEW
125 RTN

126♦LBL "I1"
127♦LBL A
128 FIX 2
129 RCL 04
130 RCL 06
131 /
132 RCL 03
133 X<>Y
134 RCL 05
135 X↑2
136 X<> T
137 CLX
138 RDN
139 XEQ "CDIV"
140 RCL 02
141 RCL 01
142 XEQ "CADD"
143 0
144 RCL 00
145 RDN
146 RDN
147 XEQ "CDIV"
148 ENTER↑
149 RDN
150 XEQ "CABS"
151 STO 15
152 "I1="
153 ARCL X
154 "⊢ A"
155 AVIEW
156 RTN

157♦LBL "COSF"
158 FIX 4
159 STO Y
160 R↑
161 X<>Y
162 /
163 STO 16
164 "COSF="
165 ARCL X
166 AVIEW
167 RTN

168♦LBL "PEL"
169 FIX 1
170 RCL 00
171 RCL 15
172 *
173 3
174 *
175 RCL 16
176 *
177 STO 13
178 "PEL="
179 ARCL X
180 AVIEW
181 STOP

182♦LBL "PV1"
183 RCL 01
184 RCL 15
185 X↑2
186 *
187 3
188 *
189 "PV1="
190 ARCL X
191 AVIEW
192 STOP

193♦LBL "PDEL"
194 CHS
195 RCL 13
196 +
197 STO 14
198 "PDEL="
199 ARCL X
200 AVIEW
201 STOP

202♦LBL "PMECH"
203 1
204 RCL 06
205 -
206 *
207 "PMECH="
208 ARCL X
209 AVIEW
210 STOP

211♦LBL "ETA"
212 RCL 13
213 /
214 FIX 5
215 "ETA="
216 ARCL X
217 AVIEW
218 STOP

219♦LBL "PV2"
220 FIX 1
221 RCL 14
222 RCL 06
223 *
224 "PV2="
225 ARCL X
226 AVIEW
227 STOP

228♦LBL "MeL"
229♦LBL B
230 FIX 2
231 RCL 08
232 RCL 06
233 /
234 RCL 01
235 +
236 X↑2
237 RCL 07
238 X↑2
239 +
240 RCL 00
241 X↑2
242 X<>Y
243 /
244 RCL 08
245 RCL 06
246 /
247 *
248 RCL 09
249 *
250 2
251 /
252 PI
253 /
254 RCL 10
255 /
256 3
257 *
258 RND
259 "MeL="
260 ARCL X
261 "⊢ NM"
262 AVIEW
263 RTN
264 GTO "I1"

265♦LBL "FNEU"
266 RCL 10
267 50
268 /
269 RCL 17
270 X<>Y
271 STO 17
272 X<>Y
273 /
274 ST* 02
275 ST* 03
276 ST* 05
277 ST* 11
278 XEQ "UeK"
279 X↑2
280 RCL 04
281 *
282 STO 08
283 RTN

284♦LBL "S"
285♦LBL C
286 "GEN=FS00"
287 AVIEW
288 STOP
289 RCL 21
290 RCL 12
291 *
292 E↑X
293 RCL 20
294 *
295 FS? 00
296 CHS
297 STO 06

298♦LBL 00
299 FIX 9
300 VIEW 06
301 XEQ "I1"
302 RCL 12
303 -
304 STO Y
305 ABS
306 1 E-3
307 X>Y?
308 GTO 01
309 RDN
310 RDN
311 RCL 21
312 *
313 CHS
314 1
315 +
316 ST* 06
317 GTO 00
318 CF 00

319♦LBL 01
320 FIX 6
321 "SN="
322 ARCL 06
323 AVIEW
324 RTN
325 FIX 1
326 1
327 RCL 06
328 -
329 RCL 11
330 *
331 "N="
332 ARCL X
333 "⊢ 1/MIN"
334 AVIEW
335 STOP
336 GTO "I1"

337♦LBL "V"
338 .1
339 STO 06
340 XEQ "I1"
341 STO 18
342 1
343 STO 06
344 XEQ "I1"
345 STO 19
346 RCL 18
347 X<>Y
348 /
349 CHS
350 1
351 +
352 1/X
353 .1
354 LN
355 *
356 STO 20
357 RCL 19
358 /
359 CHS
360 STO 21
361 RCL 20
362 E↑X
363 STO 20
364 GTO "S"
365 .END.
```

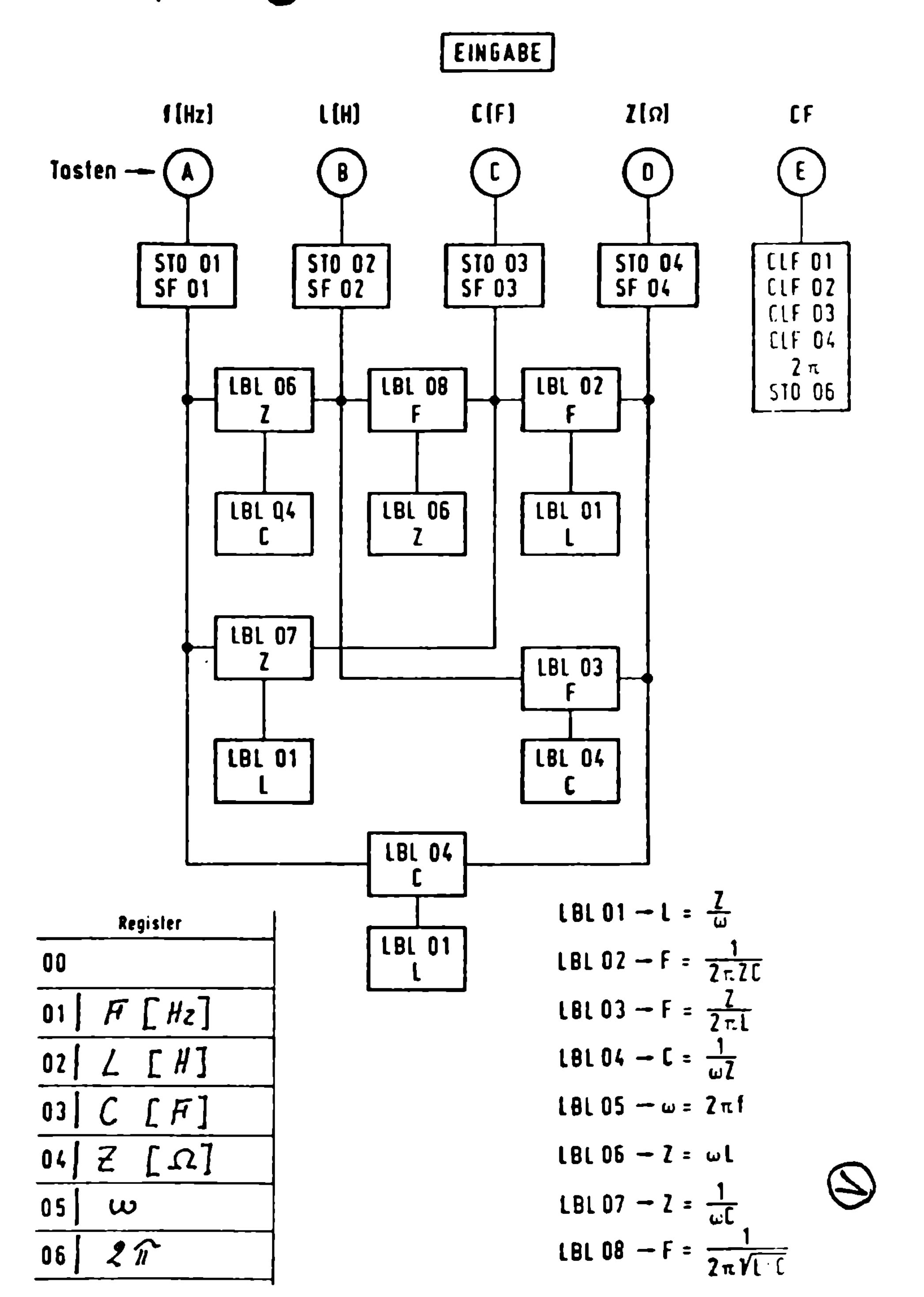
KLAUS SCHMITT
Hemsdorf

```
01◆LBL "HF"
02 SF 27
03 ENG 2
04◆LBL E
05 CF 01
06 CF 02
07 CF 03
08 CF 04
09 2
10 PI
11 *
12 STO 06
13 RTN
14◆LBL A
15 STO 01
16 SF 01
17 FS? 02
18 XEQ 06
19 FS? 03
20 XEQ 07
21 FS? 04
22 XEQ 04
23 RTN
24◆LBL B
25 STO 02
26 SF 02
27 FS? 01
28 XEQ 06
29 FS? 03
30 XEQ 08
31 FS? 04
32 XEQ 03
33 RTN
34◆LBL C
35 STO 03
36 SF 03
37 FS? 01
38 XEQ 07
39 FS? 02
40 XEQ 08
41 FS? 04
42 XEQ 02
43 RTN
44◆LBL D
45 STO 04
46 SF 04
47 FS? 01
48 XEQ 04
49 FS? 02
50 XEQ 03
51 FS? 03
52 XEQ 02
53 RTN
54◆LBL 00
55 ARCL X
56 AVIEW
57 RTN
58◆LBL 01
59 RCL 04
60 XEQ 05
61 /
62 STO 02
63 "L="
64 XEQ 00
65 RTN
66◆LBL 02
67 RCL 06
68 RCL 04
69 *
70 RCL 03
71 *
72 1/X
73 STO 01
74 "F="
75 XEQ 00
76 STOP
77 GTO 01
78◆LBL 03
79 RCL 04
80 RCL 06
81 RCL 02
82 *
83 /
84 STO 01
85 "F="
86 XEQ 00
87 STOP
88 GTO 04
89◆LBL 04
90 XEQ 05
91 RCL 04
92 *
93 1/X
94 STO 03
95 "C="
96 XEQ 00
97 STOP
98 GTO 01
99◆LBL 05
100 RCL 06
101 RCL 01
102 *
103 STO 05
104 RTN
105◆LBL 06
106 XEQ 05
107 RCL 02
108 *
109 STO 04
110 "XL="
111 XEQ 00
112 STOP
113 GTO 04
114◆LBL 07
115 XEQ 05
116 RCL 03
117 *
118 1/X
119 STO 04
120 "XC="
121 XEQ 00
122 STOP
123 GTO 01
124◆LBL 08
125 RCL 02
126 RCL 03
127 *
128 SQRT
129 RCL 06
130 *
131 1/X
132 STO 01
133 "F="
134 XEQ 00
135 STOP
136 GTO 06
137 END
```

Dieses Programm "HF-Tapete" berechnet den Zusammenhang zwischen den vier Größen F(Hz), L(H), C(F) und Z(Ω). Zunächst werden mit der Taste E alle Flags gelöscht. In der Anzeige erscheint 2 x π = 6.28. Dann werden über die Tasten ABCD zwei Größen eingegeben. Nach der 2. Eingabe wird die 3. Größe berechnet, falls die 4. Größe gewünscht wird, drücken Sie die Taste R/S.

H.HENZE

HF - Tapete



Register	
00	
01	F [Hz]
02	L [H]
03	C [F]
04	Z [Ω]
05	ω
06	2π

LBL 01 — $L = \frac{Z}{\omega}$

LBL 02 — $F = \frac{1}{2\pi ZC}$

LBL 03 — $F = \frac{Z}{2\pi L}$

LBL 04 — $C = \frac{1}{\omega Z}$

LBL 05 — $\omega = 2\pi f$

LBL 06 — $Z = \omega L$

LBL 07 — $Z = \frac{1}{\omega C}$

LBL 08 — $F = \frac{1}{2\pi\sqrt{L \cdot C}}$

H. HENZE

PROGRAMMBESCHREIBUNG

Dieses Programm Tiefpaßfilter (TPFIL) 2. Ordnung mit Einfachmitkopplung berechnet die Widerstände R_1, R_2 und C_2. Der Berechnungsgang ist folgender:

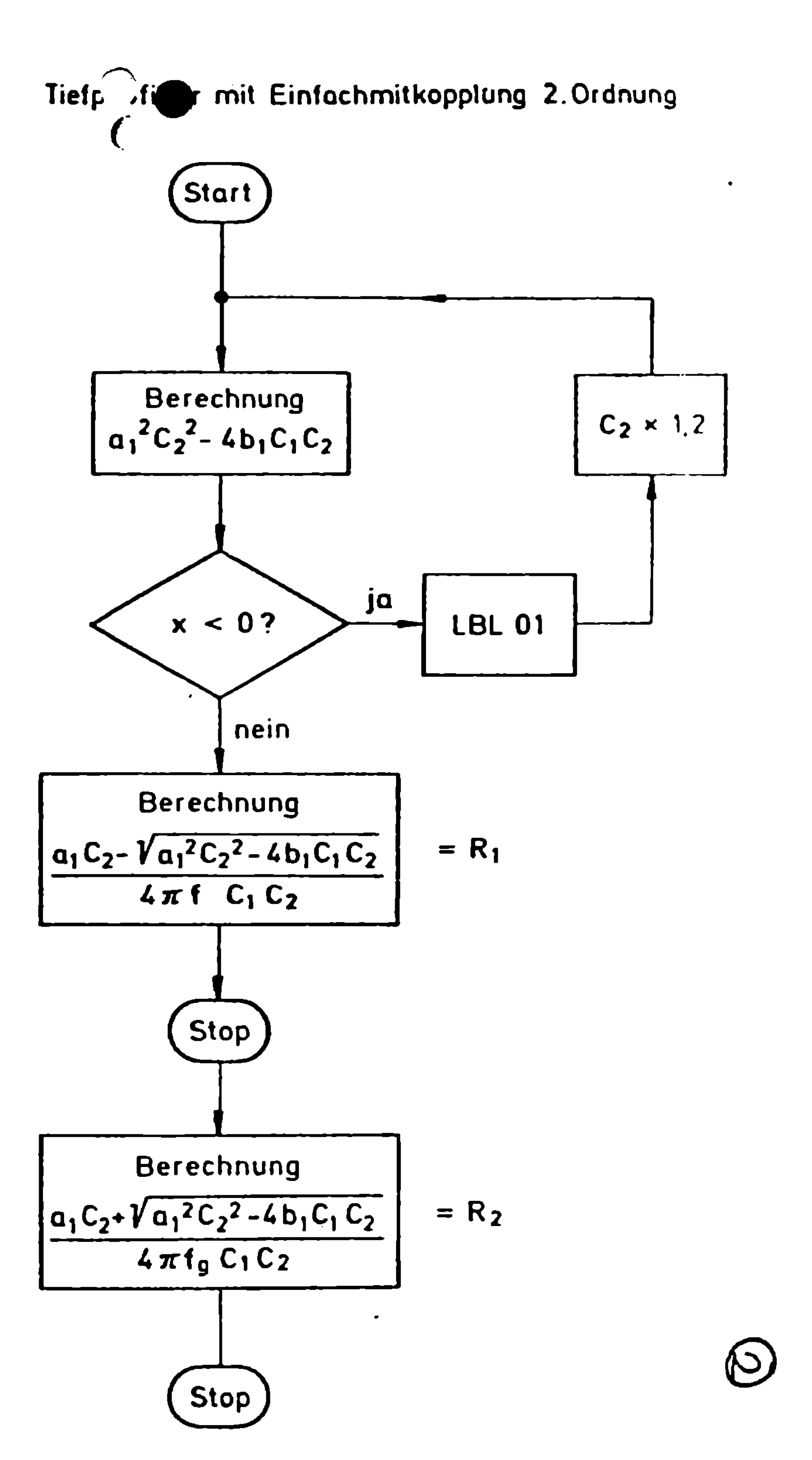
Nachdem das Programm eingegeben ist, werden die gewünschte Grenzfrequenz (f_g), die Kapazität, $C_1 = C_2$ in die Register eingegeben. Dann entnimmt man für den gewünschten Filtertyp (Bessel, Butterworth oder Tschebyscheff) die Koeffizienten a_i und b_i aus dem Buch: Halbleiterschaltungstechnik von U. Tietze u. Ch. Schenk (5. Auflage), Springer-Verlag. Diese Koeffizienten werden in Register 03 und 04 eingegeben.

Dann Taste A drücken; nun vergrößert das Programm C_2, bis der Wert unter der Wurzel größer als Null wird. Beim 1. STOP steht der Wert R_1 in der Anzeige. Taste R/S drücken: R_2 wird berechnet. Falls R_1 und R_2 nicht in der gewünschten Größenordnung liegen, muß C_1 geändert werden.

Nun ruft man C_1 und C_2 aus den Registern 01 und 02; diese Werte entnimmt man dem Lager, mißt sie mit einem Kapazitätsmeßgerät aus und läßt die genauen Werte wieder in die Rechnung einfließen. Die so gewonnenen Werte für R_1 und R_2 müssen nun ebenfalls sehr genau ausgesucht werden.

Für Filter höherer Ordnung muß dieses Spiel für jedes Teilfilter durchgeführt werden. Die hierfür erforderlichen Koeffizienten sind im genannten Buch bis zur 10. Ordnung ausgerechnet.

```
01◆LBL "TPF
IL"
02◆LBL A
03 RCL 03
04 X↑2
05 RCL 02
06 X↑2
07 *
08 4
09 RCL 04
10 *
11 RCL 01
12 *
13 RCL 02
14 *
15 -
16 X<0?
17 GTO 01
18 SQRT
19 STO 05
20 RCL 03
21 RCL 02
22 *
23 STO 06
24 X<>Y
25 -
26 4
27 PI
28 *
29 RCL 00
30 *
31 RCL 01
32 *
33 RCL 02
34 *
35 STO 07
36 /
37 STO 08
38 STOP
39 RCL 05
40 RCL 06
41 +
42 RCL 07
43 /
44 STO 09
45 STOP
46◆LBL 01
47 1.2
48 ST* 02
49 GTO A
50 END
```



Tiefp fi r mit Einfachmitkopplung 2. Ordnung
Start
Berechnung
$a_1^2 C_2^2 - 4 b_1 C_1 C_2$
$C_2 \times 1{,}2$
$x < 0$?
ja
LBL 01
nein
Berechnung
$\frac{a_1 C_2 - \sqrt{a_1^2 C_2^2 - 4 b_1 C_1 C_2}}{4 \pi f \; C_1 C_2}$
$= R_1$
Stop
Berechnung
$\frac{a_1 C_2 + \sqrt{a_1^2 C_2^2 - 4 b_1 C_1 C_2}}{4 \pi f_g \; C_1 C_2}$
$= R_2$
Stop

PGM. BEISPIELE

1) TPFIL

fg = 5 kHz	STO 00
C1 = 1 nF	STO 01
C2 = 1 nF	STO 02
a1 = 1,36	STO 03
b1 = o,62	STO 04

Ergebnis :

R1 = 15964 Ohm

R2 = 27325,31 Ohm

2) HFIL

fg = 1 kHz	STO 03
C1 = 1o nF	STO 01
C2 = 1o nF	STO 02
a1 = 1,36	STO 04
b1 = o,62	STO 05

Ergebnis :

R1 = 23,4 kOhm

R2 = 17,5 kOhm

3) PRGM ABS

T und PI Abschwächer

dB	1o,oo	STO 00
Z1	5o,oo	STO 01
Z2	5o,oo	STO 02
XEQ "ABS"		R6=71,15
		R5=96,25
		R4=96,25
SF 01 RUN		R3=35,14
		R2=25,97
		R1=25,97

dB	2o,oo	STO 00
Z1	6oo,oo	STO 01
Z2	5o,oo	STO 02
SF 01		
XEQ "ABS"		R3=136,36
		R2=-72,73
		R1=627,27
CF 01 RUN		R6=22o,oo
		R5=47,83
		R4=-412,5o

Programmbeschreibung

Das Programm Hochpaßfilter (HFIL) berechnet ein aktives Filter 2. Ordnung. Für Filter höherer Ordnung muß die Berechnung für jedes Teilfilter mit den entsprechenden Koeffizienten durchgeführt werden.

Programmausführung:

1.) Eingabe der Grenzfrequenz fg [Hz] → 03

2.) Eingabe der Kapazität C_1 und C_2 [F] → 01,02 kann zunächst gleich groß gewählt werden

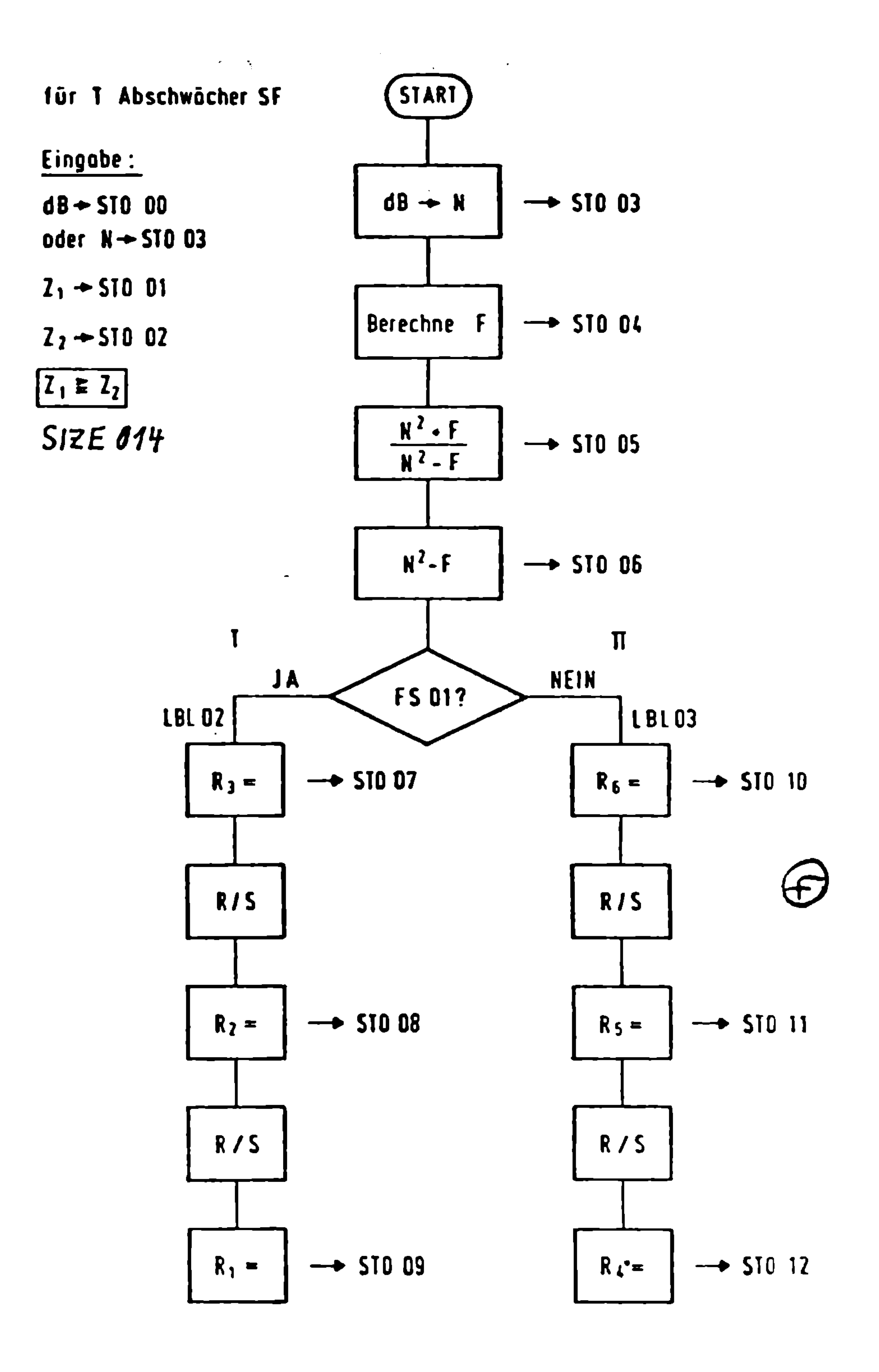
3.) Eingabe der Koeffizienten a_1 und b_1. Diese Werte, je nach Filtertyp, entnimmt man dem Buch: Halbleiterschaltungstechnik von U. Tietze und Ch. Schenk (5. Auflage), Springer-Verlag.

4.) Durch Druck auf die Taste A wird R_1 berechnet

5.) Taste R/S drücken: R_2 wird berechnet.
Falls R_1 und R_2 nicht in der für den Op. Verstärker gewünschten Größenordnung liegen, muß C_1 oder C_2 oder beide Werte geändert werden. Dann beginnt man wieder mit Punkt 4.

H. Henze

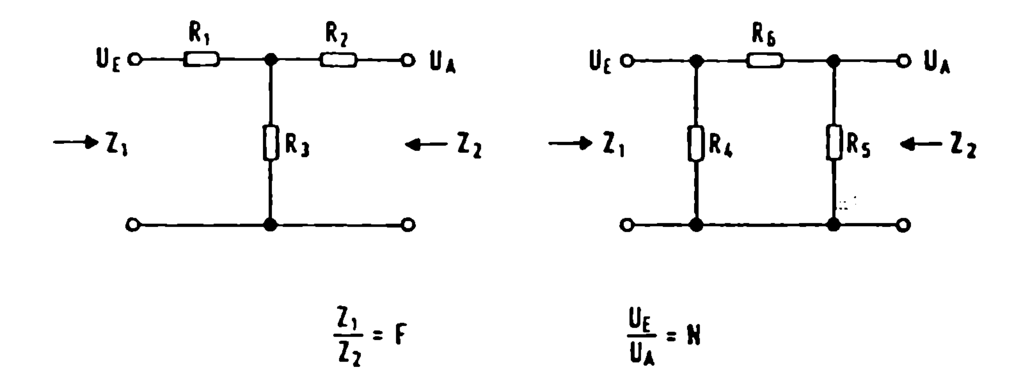
```
01◆LBL "HFI
02◆LBL A
03 2
04 PI
05 *
06 RCL 03
07 *
08 STO 06
09 RCL 04
10 *
11 1/X
12 RCL 01
13 1/X
14 RCL 02
15 1/X
16 +
17 *
18 STOP
19 RCL 04
20 RCL 05
21 RCL 06
22 *
23 /
24 RCL 01
25 RCL 02
26 +
27 1/X
28 *
29 STOP
30 END
```


für T Abschwächer SF
Eingabe:
dB → STO 00
oder N → STO 03
Z1 → STO 01
Z2 → STO 02
Z1 ≡ Z2
SIZE 014
START
dB → N
→ STO 03
Berechne F
→ STO 04
(N² + F) / (N² - F)
→ STO 05
N²-F
→ STO 06
T
Π
JA
FS 01?
NEIN
LBL 02
LBL 03
R3 =
→ STO 07
R/S
R2 =
→ STO 08
R/S
R1 =
→ STO 09
R6 =
→ STO 10
R/S
R5 =
→ STO 11
R/S
R4 =
→ STO 12

PRGM : ABS

T **ABSCHWÄCHER** Π

$Z_1 \geqq Z_2$



$$\frac{Z_1}{Z_2} = F \qquad \frac{U_E}{U_A} = N$$

$$R_3 = Z_1 \cdot \frac{2N}{N^2 - F} \qquad \frac{1}{R_6} = \frac{1}{Z_2} \cdot \frac{2N}{N^2 - F}$$

$$R_2 = \left[Z_2 \cdot \frac{N^2 + F}{N^2 - F} \right] - R_3 \qquad \frac{1}{R_5} = \left[\frac{1}{Z_2} \cdot \frac{N^2 + F}{N^2 - F} \right] - \frac{1}{R_6}$$

$$R_1 = \left[Z_1 \cdot \frac{N^2 + F}{N^2 - F} \right] - R_3 \qquad \frac{1}{R_4} = \left[\frac{1}{Z_1} \cdot \frac{N^2 + F}{N^2 - F} \right] - \frac{1}{R_6}$$

```
PRP "ABS"

01♦LBL "ABS"              53 RCL 01
                          54 1/X
02 RCL 00                 55 RCL 05
03 X=0?                   56 *
04 GTO 01                 57 RCL 13
05 20                     58 -
06 /                      59 1/X
07 10↑X                   60 STO 12
08 STO 03                 61 "R4="
                          62 ARCL X
09♦LBL 01                 63 PROMPT
10 RCL 01                 64 GTO "ABS"
11 RCL 02
12 /
13 STO 04                 65♦LBL 02
14 RCL 03                 66 RCL 03
15 X↑2                    67 2
16 RCL 04                 68 *
17 +                      69 RCL 06
18 RCL 03                 70 /
19 X↑2                    71 RCL 01
20 RCL 04                 72 *
21 -                      73 STO 07
22 STO 06                 74 "R3="
23 /                      75 ARCL X
24 STO 05                 76 PROMPT
25 FS? 01                 77 RCL 02
26 GTO 02                 78 RCL 05
                          79 *
27♦LBL 03                 80 RCL 07
28 RCL 03                 81 -
29 2                      82 STO 08
30 *                      83 "R2="
31 RCL 06                 84 ARCL X
32 /                      85 PROMPT
33 RCL 02                 86 RCL 01
34 1/X                    87 RCL 05
35 *                      88 *
36 STO 13                 89 RCL 07
37 1/X                    90 -
38 STO 10                 91 STO 09
39 "R6="                  92 "R1="
40 ARCL X                 93 ARCL X
41 PROMPT                 94 PROMPT
42 RCL 02                 95 GTO "ABS"
43 1/X
44 RCL 05                 96 END
45 *
46 RCL 13                        CAT 1
47 -                             CAT 1
48 1/X                    LBL'ABS
49 STO 11                 END
50 "R5="                  140 BYTES
51 ARCL X                 .END.
52 PROMPT                 07 BYTES
```



Hans-Günter Lütke Uphues
Bahnhofstraße 20
4401 Altenberge

(1).

HP-41 C berechnet Betriebsdämpfungsmaß und Betriebsdämpfungswinkel von Vierpolen

Dieses für den HP-41C geschriebene Programm berechnet das Betriebsdämpfungsmaß a_B und den Betriebsdämpfungswinkel b_B einer Schaltung, einer Vierpolkette, die am Eingang mit einem Generator (Innenwiderstand $\underline{Z}_i = R_i + jX_i$) und am Ausgang mit einem Lastwiderstand ($\underline{Z}_L = R_L + jX_L$) abgeschlossen ist.

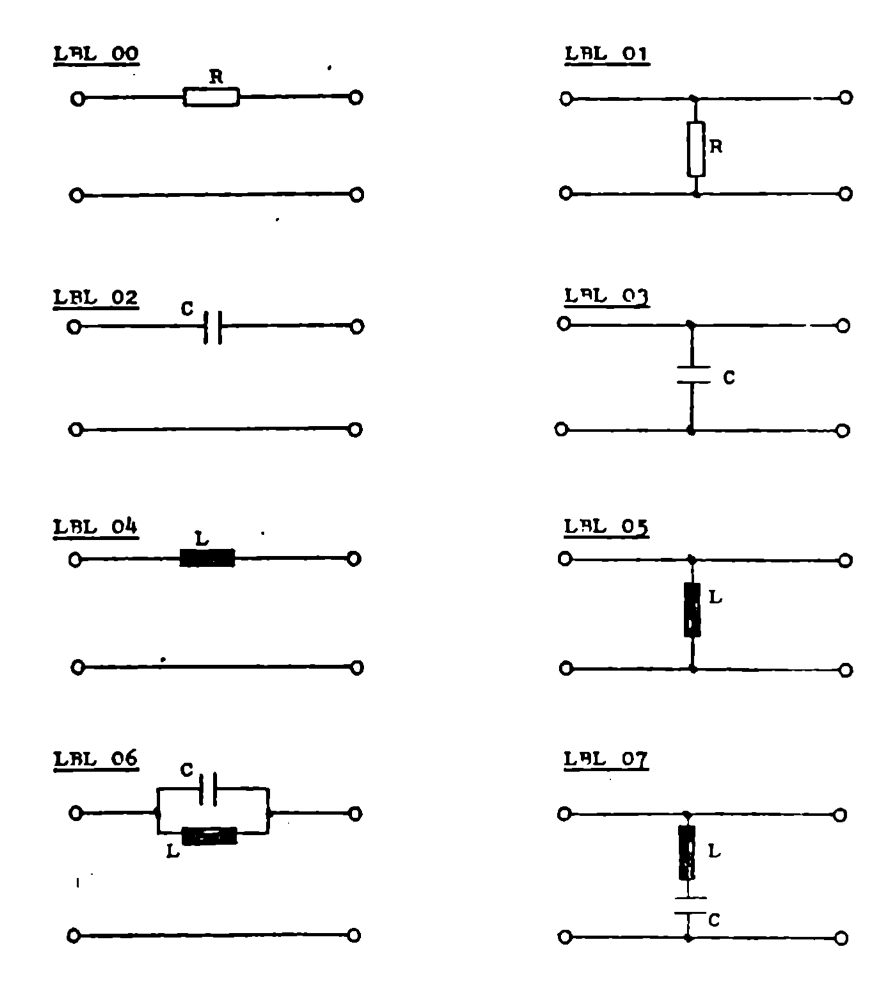
Die Vierpolkette besteht aus einer Reihenschaltung mehrere Vierpole, auch Zweitore genannt, welche alle sogenannte unvollkommene Vierpole sind, d.h.: entweder

Schaltung	Kettenmatrix
Fall a) (Längswiderstand $\underline{Z}$)	$A = \begin{vmatrix} \underline{A}_{11} = 1 & \underline{A}_{12} = \underline{Z} \\ \underline{A}_{21} = 0 & \underline{A}_{22} = 1 \end{vmatrix}$
	oder
Fall b) (Querwiderstand $\underline{Z}$)	$A = \begin{vmatrix} \underline{A}_{11} = 1 & \underline{A}_{12} = 0 \\ \underline{A}_{21} = \frac{1}{\underline{Z}} & \underline{A}_{22} = 1 \end{vmatrix}$

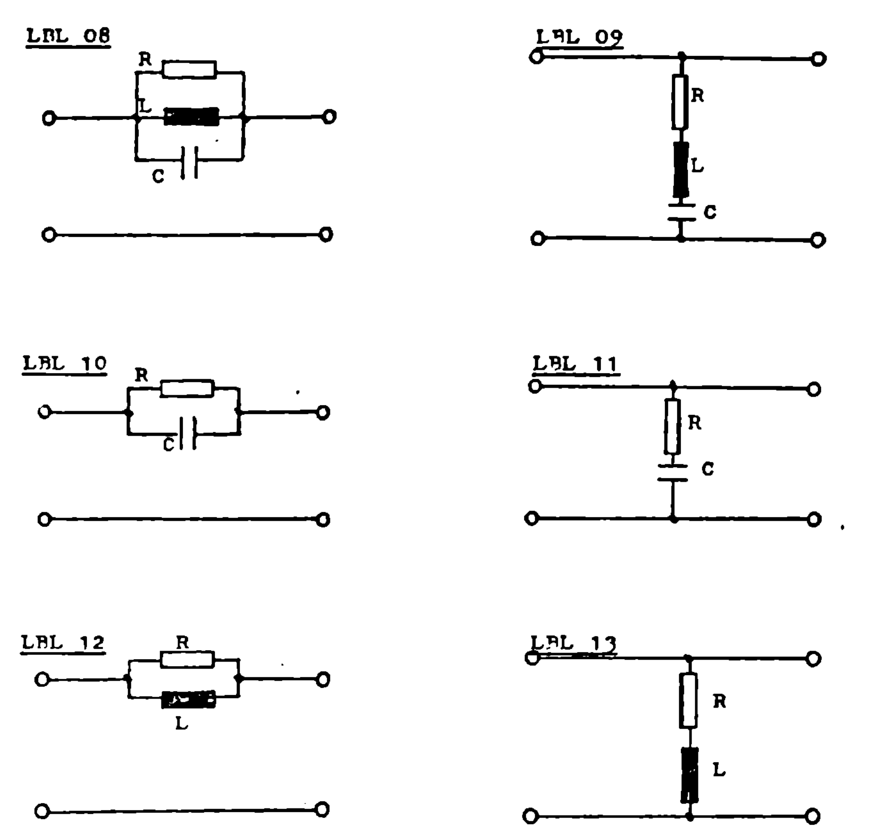
Werden mehrere Vierpole in Reihe geschaltet, so müssen die einzelnen Kettenmatrizen multipliziert werden, was nach den Regeln der Matrizenrechnung geschieht. Da es sich hier um komplexe Kettenparameter $\underline{A}_{11}$, $\underline{A}_{12}$, $\underline{A}_{21}$

und $\underline{A}_{22}$ h[illegible] nehmen diese Berechnungen (Addition, Multiplikation, Division und Wurzelziehen) einen nicht unerheblichen Teil des Programms und der Rechenzeit ein.

Mit dem hier vorgestellten Programm lassen sich folgende unvollkommene Vierpole verarbeiten. Weitere unvollkommene, aber natürlich auch vollkommene, Vierpole können leicht dem Programm hinzugefügt werden oder anstelle eines der hier aufgezählten Vierpole in das Programm aufgenommen werden. Für diese eventuellen Ergänzungen stehen die Label 23, 24, ... noch zur Verfügung.







Die Zahlenangaben bei den hier aufgeführten Vierpolen geben die Nummer des Labels an, unter dem der entsprechende Vierpol im Programm gefunden wird um dessen Kettenmatrix aufzustellen.

Theorie und Formeln:

Das Betriebsdämpfungsmaß a_B und der Betriebsdämpfungswinkel werden in diesem Programm aus dem komplexen Betriebsdämpfungsfaktor $\underline{D}_B$ und dem komplexen Betriebsdämpfungsmaß $\underline{g}_B$ wie folgt berechnet:

$$\underline{D}_B = e^{\underline{g}_B}$$

Hierin ist

$$\underline{g}_B = a_B + jb_B$$

$$= \ln \frac{1}{2} \left(\underline{A}_{11}\sqrt{\frac{\underline{Z}_L}{\underline{Z}_1}} + \underline{A}_{12}\sqrt{\frac{1}{\underline{Z}_1\underline{Z}_L}} + \underline{A}_{21}\sqrt{\underline{Z}_1\underline{Z}_L} + \underline{A}_{22}\sqrt{\frac{\underline{Z}_1}{\underline{Z}_L}} \right)$$

$\underline{D}_B$ = komplexer Betriebsdämpfungsfaktor

$\underline{g}_B$ = komplexes Betriebsdämpfungsmaß

a_B = Betriebsdämpfungsmaß

b_B = Betriebsdämpfungswinkel oder Betriebsphasenmaß

Zur Programmbedienung:

Die Reihenfolge der Vierpole muß im Anschluß an LBL A zusätzlich zu dem bestehenden Programm programmiert werden, und zwar vom Eingang zum Ausgang der Vierpolkette.

Bei der Werteeingabe für die Schaltelemente ist die Reihenfolge R, L, C für alle Vierpole gleich. Besteht ein Vierpol aus nur einem Schaltelement muß natürlich nur ein Schaltelement dem Rechner eingetippt werden. Zur Eingabe dieser Werte, sowie der Werte des Generatorinnenwiderstandes und des Lastwiderstandes wird das Alpha-Label "EING" wie Eingabe verwendet und für den Berechnungsstart das Alpha-Label "START". Nach dessen Aufruf müssen noch Anfangsfrequenz f, Frequenzschrittweite Δf und die Endfrequenz f_{max} eingegeben werden. Danach startet das Programm mit der Berechnung und endet mit der Anzeige des Wertes $f = f_{max} + \Delta f$.
Es wurde diese Eingaberoutine gewählt, um Berechnungen mit anderen Frequenzen an der selben Schaltung leicht vornehmen zu können, wofür einfach nur wieder das Label "START" aufgerufen werden muß.

Da bei Schaltungen mit vielen Bauelementen eine Fehleingabe vorkommen kann, ist auch eine Korrekturroutine ins Programm aufgenommen worden. Wenn man z.B. nach Eingabe des 10.

Schaltelementes feststellt, daß das 2. Bauelement mit einem falschen Wert eingetippt worden ist, kann man hier die Eingaberoutine unterbrechen, die Nr. des zu korrigierenden Bauelementes (hier 2) in die Anzeige tippen und mit XEQ ^{T}KOR die Korrekturroutine aufrufen, wonach in der Anzeige wieder R,L,C?n (hier n = 2) erscheint. Jetzt tippt man den richtigen Wert ein, drückt R/S und man kann nach Anzeige von R,L,C?n an der Stelle der Eingaberoutine fortfahren, wo man sie unterbrochen hatte.

Register- und Label-Belegung des Programms:

A) Registerbelegung:

R00 ... R19: für komplexe Matrizen
R20: Frequenz f
R21: Frequenzschrittweite Δf
R22: 2 x PI x f = ω
R23: verschiedenes
R24: "
R25: Endfrequenz f_{max}
R26: Zähler
R27: reller Innenwiderstand des Generators R_i
R28: imaginärer Innenwiderstand des Generators jX_i
R29: reller Lastwiderstand R_L
R30: imaginärer Lastwiderstand jX_L
R31: Inneninduktivität L_i bzw. Innenkapazität $-C_i$ des G.
R32: Lastinduktivität L_L bzw. Lastkapazität $-C_L$
R33 .....: Werte der Schaltelemente, beginnend mit n = 1

B) Labelbelegung:

LBL 00 ... 13: für Kettenmatrizenaufstellung eines der angegebenen Vierpole
LBL 14: Wurzelziehen aus einer komplexen Zahl (Unterprogramm)

LBL 15: komplexe Multiplikation (Unterprogramm)

LBL 16: Matrizenaustausch

LBL 17: Schleife zur Werte-Eingabe der Bauelemente

LBL 18 und 21: Schleife zur Berechnung des Betriebsdämpfungsmaßes a_B und des Betriebsdämpfungswinkels b_B in Frequenzschritten von Δf

LBL 19: komplexe Division (Unterprogramm)

LBL 20: Sprungadresse für komplexe Division in das Unterprogramm der komplexen Multiplikation

LBL 22: Komplexe Matrizenmultiplikation

LBL A: Label zum Auffinden der Pgm-Zeile, ab der die Kettenmatrix-Unterprogramme aufgerufen werden

LBL ᵀEING: Eingabe des Generatorinnenwiderstandes $\underline{Z}_i$, des Lastwiderstandes $\underline{Z}_L$, sowie der Werte der Schaltelemente

LBL ᵀSTART: Eingabe von f, Δf, f_{max} und Programmstart

LBL ᵀKOR: Korrektur einer falschen Eingabe eines Schaltelementes

Schlußbemerkungen:

Soll das Programm in dieser Form verwendet werden, ist ein Drucker notwendig. Steht dieser aber nicht zur Verfügung, müssen die Druckbefehle PRA (= Drucke Inhalt des Alpha-Registers) gelöscht werden. Bei der Programmzeile 102 kann zur Anzeige der Frequenz der PRA-Befehl durch den AVIEW-Befehl und bei den Programmzeilen 193 und 197 durch je einen PROMPT-Befehl ersetzt werden, um a_B bzw. b_B anzuzeigen und das Programm abzustoppen. Sind diese Änderungen vorgenommen, muß nach jedem berechneten Wert R/S zur Fortsetzung der Berechnung gedrückt werden.

Das gesamte Programm (1022 Bytes) kann auf 10 Magnetkartenseiten dauerhaft gespeichert werden.

Literaturhinweise:

E. Philippow: Grundlagen der Elektrotechnik

Brühl/Jansen/Vogt: Nachrichtenübertragungstechnik I

Stefan Hamerli: Wobbelkurven ohne Meßgerät ermittelt: Elektronik 5/1978

Programmbedienung:

Schritt	Prozedur	Eingabe	Taste	Anzeige
1	Eingabe der Vierpolkette:		GTO A PRGM XEQ XX ⋮ PRGM	 123 LBL A XEQ XX ⋮
2	Eingabe des Generatorinnenwiderstandes, des Lastwiderstandes und die Werte der Bauelemente Vorbereitung **Achtung:** **Falls Innenkapazität oder Lastkapazität vorhanden, müssen diese mit negativem Vorzeichen eingegeben werden.** Wenn alle Schaltelemente eingegeben wurden: weiter mit 4	 R_i L_i bzw. $-C_i$ R_L L_L bzw. $-C_L$ R,L oder C_1 R,L oder C_2 ⋮	XEQ ^{T}EING R/S R/S R/S R/S R/S R/S ⋮	RI? LI,-CI? RL? LL,-CL? R,L,C?1 R,L,C?2 ⋮
3	Korrektur einer R,L,C-Eingabe (n-tes Bauelement) weiter bei 2	n R,L oder C_n	XEQ ^{T}KOR R/S	R,L,C?n
4	Frequenzeingaben: Startfrequenz Frequenzschrittweite Endfrequenz Berechnungsstart. Rechnung endet mit	 f Δf f_{max}	XEQ ^{T}START R/S R/S R/S	F? dF? F MAX? $f_{max} + \Delta f$

```
01♦LBL "EING"
ENG 9  DEG  "RI?"
PROMPT  "⊢= "  ARCL X
PRA  STO 27  "LI,-CI?"
PROMPT  "⊢= "  ARCL X
PRA  STO 31  "RL?"
PROMPT  "⊢= "  ARCL X
PRA  STO 29  "LL,-CL?"
PROMPT  "⊢= "  ARCL X
PRA  STO 32  ADV  ADV
33  STO 26  1  STO 25
CF 29

35♦LBL 17
"R,L,C?"  FIX 0  ARCL 25
ENG 9  PROMPT  "⊢= "
ARCL X  PRA  STO IND 26
1  ST+ 25  ST+ 26
GTO 17
49♦LBL "START"
SF 29  ADV  "F?"  PROMPT
"⊢= "  ARCL X  "⊢ HZ"
PRA  "dF?"  PROMPT
"⊢= "  ARCL X  "⊢ HZ"
PRA  STO 21  -  STO 20
"F MAX?"  PROMPT  "⊢= "
ARCL X  "⊢ HZ"  PRA
STO 25  ADV  ADV

98♦LBL 21
"F= "  ARCL X  "⊢ HZ"
PRA  2  PI  *  *  STO 22
RCL 31  X<>Y  *  STO 28
LASTX  RCL 32  *  STO 30
RCL 26  INT  1 E3  /  33
+  STO 26

123♦LBL A
RCL 30  X<0?  1/X
STO 30  RCL 29  RCL 28
X<0?  1/X  STO 28
RCL 27  XEQ 19  XEQ 14
STO 08  X<>Y  STO 09
X<>Y  RCL 01  RCL 00
XEQ 15  STO 00  X<>Y
STO 01  RCL 28  RCL 27
RCL 30  RCL 29  XEQ 15
XEQ 14  STO 10  X<>Y
STO 11  X<>Y  RCL 03
RCL 02  RDN  RDN  XEQ 19
ST+ 00  X<>Y  ST+ 01
RCL 05  RCL 04  RCL 11
RCL 10  XEQ 15  ST+ 00
X<>Y  ST+ 01  RCL 07
RCL 06  RCL 09  RCL 08
XEQ 19  ST+ 00  X<>Y
ST+ 01  RCL 01  2  /
RCL 00  2  /  R-P  LOG
20  *  "aB= "  ARCL X
"⊢ dB"  PRA  "bB= "
ARCL Y  "⊢ GRAD"  PRA
GTO 18

199♦LBL 00
RCL IND 26  STO 16  0
STO 17  STO 18  STO 19
GTO 22

207♦LBL 01
RCL IND 26  1/X  STO 18
0  STO 19  STO 16
STO 17  GTO 22

216♦LBL 02
RCL IND 26  RCL 22  *
1/X  CHS  STO 17  0
STO 16  STO 18  STO 19
GTO 22

228♦LBL 03
RCL IND 26  RCL 22  *
STO 19  0  STO 18
STO 16  STO 17  GTO 22

238♦LBL 04
RCL IND 26  RCL 22  *
STO 17  0  STO 16
STO 18  STO 19  GTO 22

248♦LBL 05
RCL IND 26  RCL 22  *
1/X  CHS  STO 19  0
STO 18  STO 17  STO 16
GTO 22

260♦LBL 06
RCL IND 26  RCL 22  *
1/X  ISG 26  RCL IND 26
RCL 22  *  -  1/X
STO 17  0  STO 16
STO 18  STO 19  GTO 22

277♦LBL 07
RCL IND 26  ISG 26
RCL IND 26  RCL 22  *
1/X  X<>Y  RCL 22  *  -
1/X  STO 19  0  STO 18
STO 17  STO 16  GTO 22

295♦LBL 08
RCL IND 26  1/X  STO 23
X↑2  ISG 26  RCL IND 26
RCL 22  *  1/X  CHS
ISG 26  RCL IND 26
RCL 22  *  +  STO 24
X↑2  +  1/X  ENTER↑
ENTER↑  RCL 23  *
STO 16  X<>Y  RCL 24  *
CHS  STO 17  0  STO 18
STO 19  GTO 22

329♦LBL 09
RCL IND 26  STO 23  X↑2
ISG 26  RCL IND 26
RCL 22  *  ISG 26
RCL IND 26  RCL 22  *
1/X  -  STO 24  X↑2  +
1/X  ENTER↑  ENTER↑
RCL 23  *  STO 18  X<>Y
RCL 24  *  CHS  STO 19
0  STO 16  STO 17
GTO 22

361♦LBL 10
RCL IND 26  STO 23
ISG 26  RCL IND 26
RCL 22  *  *  STO 24
X↑2  1  +  1/X  ENTER↑
ENTER↑  RCL 23  *
STO 16  X<>Y  RCL 24  *
CHS  RCL 23  *  STO 17
0  STO 18  STO 19
GTO 22

390♦LBL 11
RCL IND 26  STO 23  X↑2
ISG 26  RCL IND 26
RCL 22  *  1/X  STO 24
X↑2  +  1/X  ENTER↑
ENTER↑  RCL 23  *
STO 18  X<>Y  RCL 24  *
STO 19  0  STO 16
STO 17  GTO 22

416♦LBL 12
RCL IND 26  STO 23
ISG 26  RCL IND 26
RCL 22  *  STO 24  /
ENTER↑  1/X  +  1/X
ENTER↑  ENTER↑  RCL 24
*  STO 16  X<>Y  RCL 23
*  STO 17  0  STO 18
STO 19  GTO 22

442♦LBL 13
RCL IND 26  STO 23  X↑2
ISG 26  RCL IND 26
RCL 22  *  STO 24  X↑2
+  1/X  ENTER↑  ENTER↑
RCL 23  *  STO 18  X<>Y
RCL 24  *  CHS  STO 19
0  STO 16  STO 17

467♦LBL 22
RCL 01  STO 09  RCL 00
STO 08  RCL 17  RCL 16
XEQ 15  STO 10  X<>Y
STO 11  RCL 03  ST+ 11
RCL 02  ST+ 10  RCL 19
RCL 18  XEQ 15  ST+ 08
X<>Y  ST+ 09  RCL 05
STO 13  RCL 04  STO 12
RCL 17  RCL 16  XEQ 15
STO 14  X<>Y  STO 15
RCL 07  ST+ 15  RCL 06
ST+ 14  RCL 19  RCL 18
XEQ 15  ST+ 12  X<>Y
ST+ 13  15,007  STO 23
7  STO 24

512♦LBL 16
RCL IND 23  STO IND 24
1  ST- 24  DSE 23
GTO 16  ISG 26  RTN

521♦LBL 19
R-P  1/X  X<>Y  CHS
GTO 20

527♦LBL 15
R-P  X<>Y

530♦LBL 20
RDN  RDN  R-P  R↑  *
RDN  +  R↑  P-R  RTN

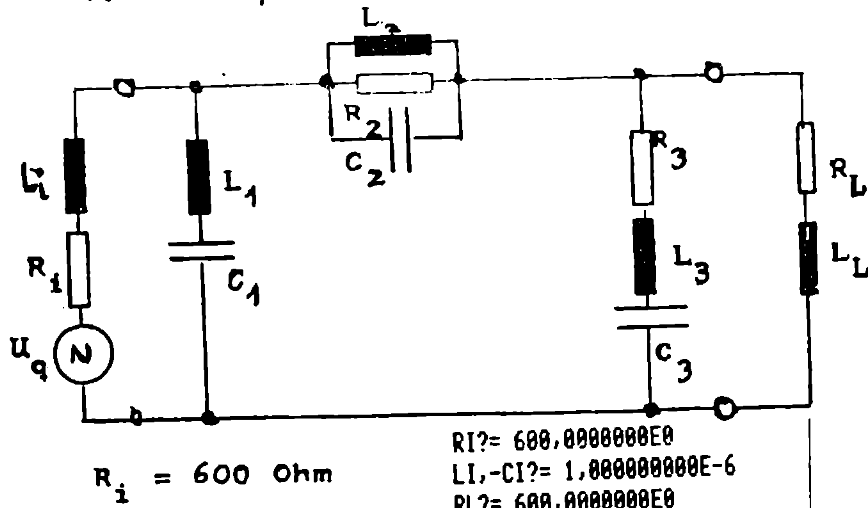
541♦LBL 14
R-P  SQRT  X<>Y  2  /
ENTER↑  SIN  RCL Z  *
X<>Y  COS  RCL Z  *  RTN

556♦LBL "KOR"
FIX 0  "KOR.: "  ARCL X
PRA  "R,L,C?"  ARCL X
32  +  ENG 9  PROMPT
"⊢= "  ARCL X  PRA
STO IND Y  ADV  GTO 17
END

76♦LBL 18
ADV  1  STO 00  STO 06
0  STO 01  STO 02
STO 03  STO 04  STO 05
STO 07  RCL 21  ST+ 20
RCL 25  RCL 20  X<=Y?
GTO 21  ADV  ADV  ADV
STOP
```

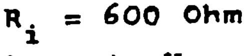
BEISPIELAUFGABEN

1.: Bandsperre in PI- Schaltung:



(11)

R_i = 600 Ohm
L_i = 1 µH
R_L = 600 Ohm
L_L = 0,5 mH



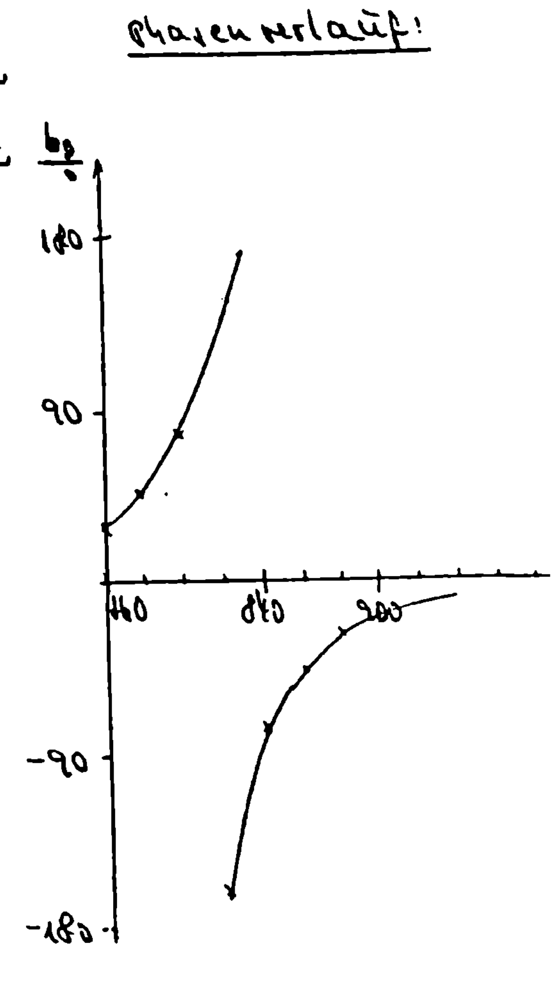
L_1 = 2,38 H
C_1 = 15,79 nF
R_2 = 1 E kOhm
L_2 = 11,37 mH
C_2 = 3,32 µF
R_3 = 2 Ohm
L_3 = 2,387 H
C_3 = 15,79 nF

```
RI?= 600,0000000E0
LI,-CI?= 1,000000000E-6
RL?= 600,0000000E0
LL,-CL?= 500,0000000E-6

R,L,C?1= 2,387000000E0
R,L,C?2= 15,79000000E-9
R,L,C?3= 1,000000000E3
R,L,C?4= 11,37000000E-3
R,L,C?5= 3,320000000E-6
R,L,C?6= 2,000000000E0
R,L,C?7= 2,387000000E0
R,L,C?8= 15,79000000E-9

F?= 760,0000000E0 HZ
dF?= 20,00000000E0 HZ
F MAX?= 900,0000000E0 HZ
```

Phasenverlauf:



```
1.: BANDSPERRE
--------------

LIST 005

123*LBL A
124 XEQ 07
125 XEQ 08
126 XEQ 09
127 RCL 30
```

```
F= 760,0000000E0 HZ
aB= 995,9072418E-3 dB
bB= 35,13387186E0 GRAD

F= 780,0000000E0 HZ
aB= 2,090977686E0 dB
bB= 50,62148685E0 GRAD

F= 800,0000000E0 HZ
aB= 6,086231534E0 dB
bB= 84,60210087E0 GRAD

F= 820,0000000E0 HZ
aB= 77,23240972E0 dB
bB= -162,3024545E0 GRAD

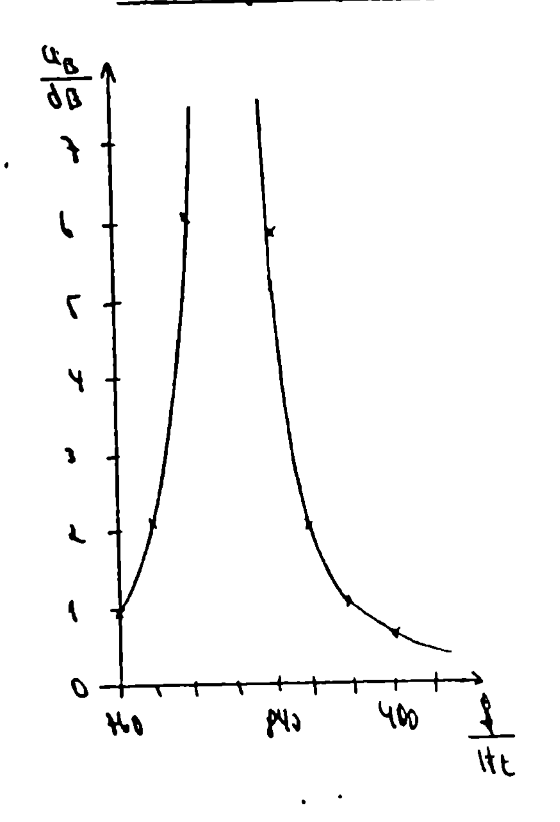
F= 840,0000000E0 HZ
aB= 5,908897432E0 dB
bB= -84,90550786E0 GRAD

F= 860,0000000E0 HZ
aB= 2,148621272E0 dB
bB= -51,93943582E0 GRAD

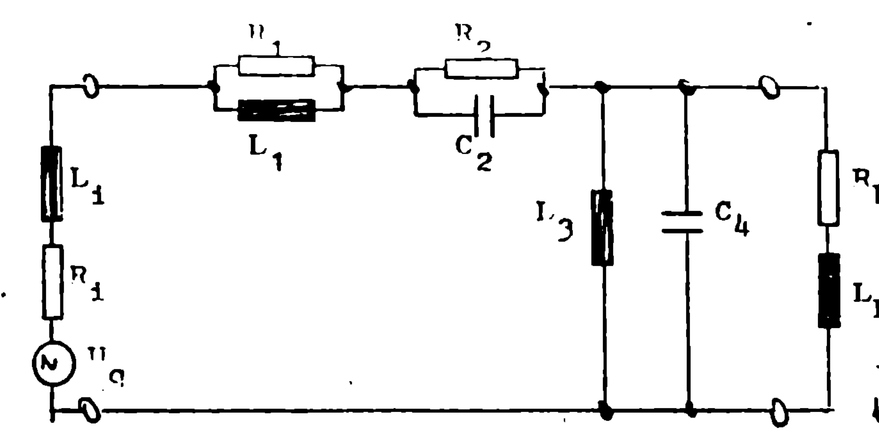
F= 880,0000000E0 HZ
aB= 1,089591769E0 dB
bB= -37,06710984E0 GRAD

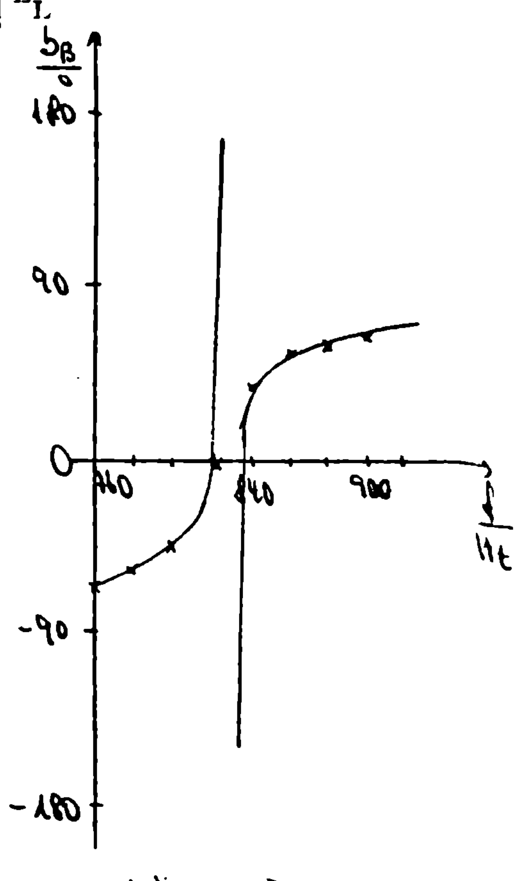
F= 900,0000000E0 HZ
aB= 656,5725332E-3 dB
bB= -28,78841355E0 GRAD
```

Dämpfungs-Verlauf



2.: Bandpaß:





R_i = 600 Ohm
L_i = 1 uH
R_L = 600 Ohm
L_L = 0,5 mH

R_1 = 10 kOhm
L_1 = 2,387 H
R_2 = 10 kOhm
C_2 = 15,79 nF
L_3 = 5,68 mH
C_4 = 6,63 uF

```
RI?= 600,00000000E0
LI,-CI?= 1,000000000E-6
RL?= 600,00000000E0
LL,-CL?= 500,0000000E-6

R,L,C?1= 10,00000000E3
R,L,C?2= 2,38700000E0
R,L,C?3= 10,00000000E3
R,L,C?4= 15,97600000E-9
R,L,C?5= 5,680000000E-3
R,L,C?6= 6,630000000E-6

F?= 760,00000000E0 HZ
dF?= 20,00000000E0 HZ
F MAX?= 900,00000000E0 HZ
```

2.: BANDPASS

```
LIST 006

123+LBL A
124 XEQ 12
125 XEQ 10
126 XEQ 05
127 XEQ 03
128 RCL 30
```

```
F= 760,00000000E0 HZ
aB= 30,77797780E0 dB
bB= -70,76079148E0 GRAD

F= 780,00000000E0 HZ
aB= 27,69521014E0 dB
bB= -62,56698329E0 GRAD

F= 800,00000000E0 HZ
aB= 23,72786074E0 dB
bB= -44,05666164E0 GRAD

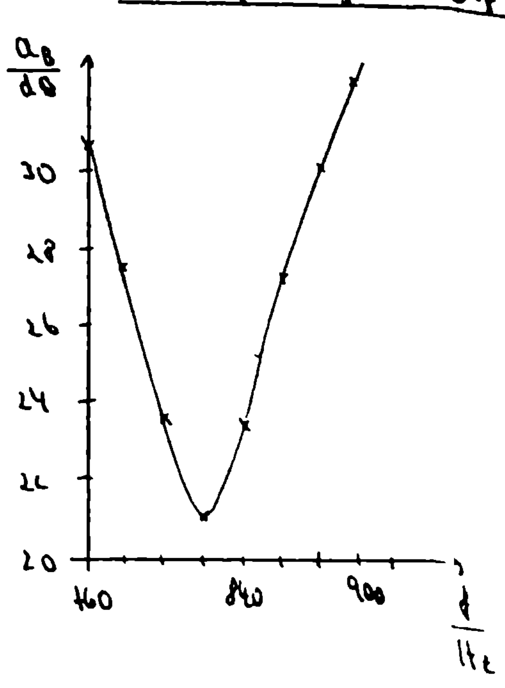
F= 820,00000000E0 HZ
aB= 20,81697244E0 dB
bB= -551,03298945E-3 GRAD

F= 840,00000000E0 HZ
aB= 23,53203216E0 dB
bB= 42,85409592E0 GRAD

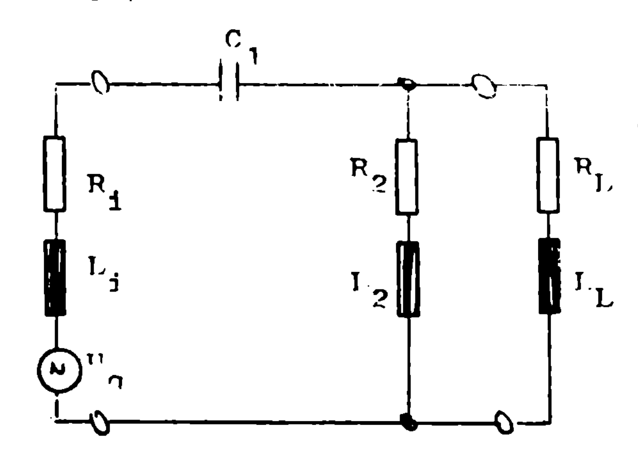
F= 860,00000000E0 HZ
aB= 27,28356704E0 dB
bB= 61,31158556E0 GRAD

F= 880,00000000E0 HZ
aB= 30,14967348E0 dB
bB= 69,50731768E0 GRAD

F= 900,00000000E0 HZ
```



3.: Hochpaß:



R_i = 600 Ohm
L_i = 1 uF
R_L = 600 Ohm
L_L = 0,5 mH

C_1 = 331,6 nF
R_2 = 1 Ohm
L_2 = 119,4 mH

```
RI?= 600,0000000E0
LI,-CI?= 1,000000000E-6
RL?= 600,0000000E0
LL,-CL?= 500,0000000E-6

R,L,C?1= 331,6000000E-9
R,L,C?2= 1,000000000E0
R,L,C?3= 119,4000000E-3

F?= 100,0000000E0 HZ
dF?= 100,0000000E0 HZ
F MAX?= 800,0000000E0 HZ
```

```
3. HOCHPASS
------------

        LIST 004

123*LBL P
124 XEQ 02
125 XEQ 13
126 RCL 30
```

```
F= 100,0000000E0 HZ
aB= 30,11876612E0 dB
bB= -164,7496392E0 GRAD

F= 200,0000000E0 HZ
aB= 18,14406741E0 dB
bB= -149,8476513E0 GRAD

F= 300,0000000E0 HZ
aB= 11,36763466E0 dB
bB= -133,5061140E0 GRAD

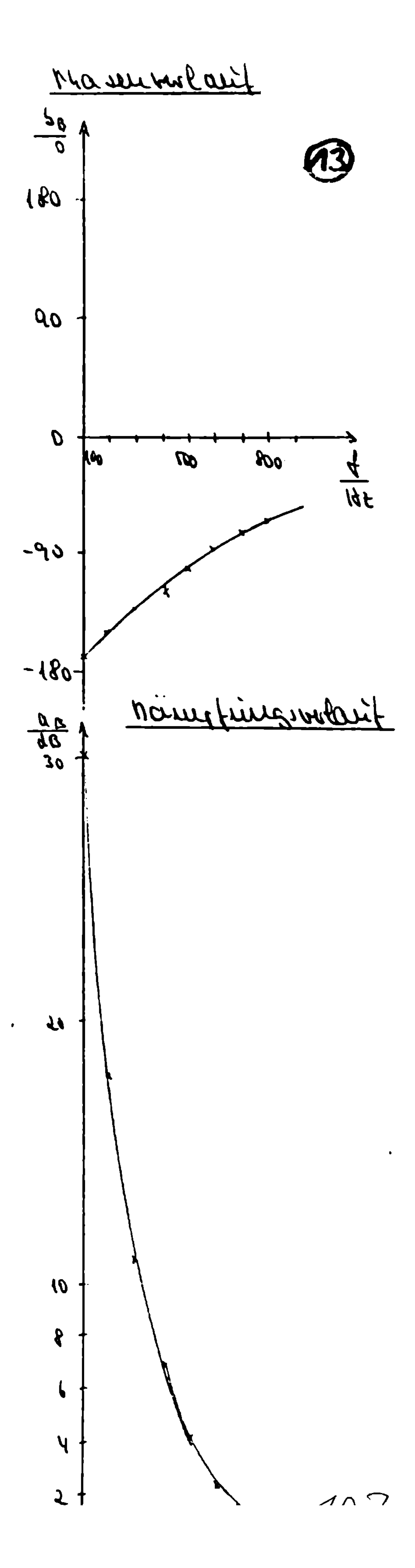
F= 400,0000000E0 HZ
aB= 7,011700324E0 dB
bB= -116,3758989E0 GRAD

F= 500,0000000E0 HZ
aB= 4,236170816E0 dB
bB= -99,80677770E0 GRAD

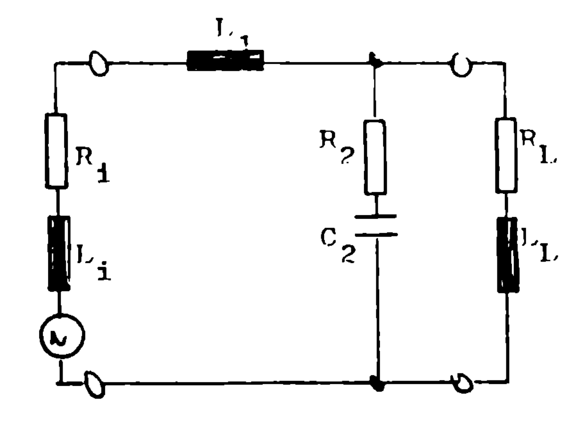
F= 600,0000000E0 HZ
aB= 2,549940534E0 dB
bB= -85,17635682E0 GRAD

F= 700,0000000E0 HZ
aB= 1,560937166E0 dB
bB= -73,08328828E0 GRAD

F= 800,0000000E0 HZ
aB= 984,3541252E-3 dB
bB= -63,42716126E0 GRAD
```



4.: Tiefpaß:



R_i = 600 Ohm
L_i = 1 μH
R_L = 600 Ohm
L_L = 0,5 mH

L_1 = 119,4 mH
R_2 = 1 Ohm
C_2 = 331,6 nF

```
RI?= 600,0000000E0
LI,-CI?= 1,000000000E-6
RL?= 600,0000000E0
LL,-CL?= 500,0000000E-6

R,L,C?1= 119,4000000E-3
R,L,C?2= 1,000000000E0
R,L,C?3= 331,6000000E-9

F?= 500,0000000E0 HZ
dF?= 500,0000000E0 HZ
F MAX?= 3,000000000E3 HZ
```

4.: TIEFPASS

```
LIST 004

123•LBL A
124 XEQ 04
125 XEQ 11
126 RCL 30
```

```
F= 500,0000000E0 HZ
aB= 164,6172960E-3 dB
bB= 37,83046950E0 GRAD

F= 1,000000000E3 HZ
aB= 2,069127772E0 dB
bB= 80,02734771E0 GRAD

F= 1,500000000E3 HZ
aB= 6,109245464E0 dB
bB= 111,9537371E0 GRAD

F= 2,000000000E3 HZ
aB= 10,30771747E0 dB
bB= 130,3438838E0 GRAD

F= 2,500000000E3 HZ
aB= 13,93691592E0 dB
bB= 141,1804234E0 GRAD

F= 3,000000000E3 HZ
aB= 17,01159854E0 dB
bB= 148,1631204E0 GRAD
```

Phasenverlauf

(14)

Dämpfungsverlauf

5.: Schwingkreis:

R_i = 600 Ohm
L_i = 1 uH
R_L = 600 Ohm
L_L = 0,5 mH

R_1 = 100 Ohm
L_2 = 2,41 H
C_2 = 10,5 nF
R_3 = 100 kOhm

```
RI?= 600,0000000E0
LI,-CI?= 1,000000000E-6
RL?= 600,0000000E0
LL,-CL?= 500,0000000E-6

R,L,C?1= 100,0000000E0
R,L,C?2= 2,410000000E0
R,L,C?3= 10,50000000E-9
R,L,C?4= 100,0000000E3

F?= 700,0000000E0 HZ
dF?= 100,0000000E0 HZ
F MAX?= 1,300000000E3 HZ
```

Phasenverlauf

$\frac{b_B}{°}$ — 90 — 0 — -90

700 — 900 — 1100 — 1300 $\frac{f}{Hz}$

15

5.: SCHWINGKREIS

```
LIST 005

123+LBL A
124 XEQ 00
125 XEQ 06
126 XEQ 01
127 RCL 30
```

```
F= 700,0000000E0 HZ
aB= 24,8322755E0 dB
bB= 86,3237112E0 GRAD

F= 800,0000000E0 HZ
aB= 28,9996844E0 dB
bB= 87,6711162E0 GRAD

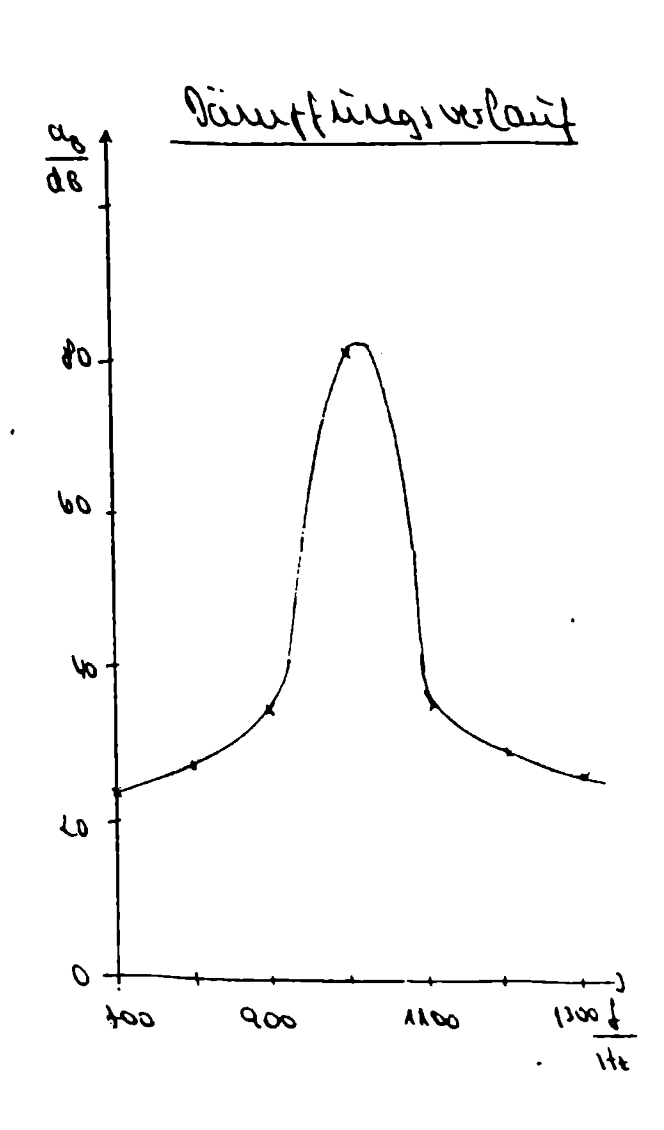
F= 900,0000000E0 HZ
aB= 35,5466732E0 dB
bB= 88,8265085E0 GRAD

F= 1,000000000E3 HZ
aB= 82,0840139E0 dB
bB= 69,8465919E0 GRAD

F= 1,100000000E3 HZ
aB= 36,5065656E0 dB
bB= -89,2323104E0 GRAD

F= 1,200000000E3 HZ
aB= 30,8186610E0 dB
bB= -88,3956757E0 GRAD

F= 1,300000000E3 HZ
aB= 27,6630,1156E0 dB
bB= -87,5972187E0 GRAD
```



Name: Widerstandsbestimmung

Beschreibung: Das Programm bestimmt den Ohm-Wert und die Toleranz von Widerständen, die mit einem 4 oder 5-stelligen Farbcode gekennzeichnet sind.

Entwicklung: Raimund Berg, Hagener Str. 2oo, 591o Kreuztal 6

Register: 13

Speicher: STO oo bis STO o2

Status: ENG 2, CF 29

Ausführung:

1 Programm eingeben
2 Tastenfeld-Maske auflegen
3 ersten Wert mit XEQ "OHM" starten, Farben der Reihe nach eintippen
4 die nächsten Werte können mit R/S gestartet werden

Tastenfeld-Maske: über die Ziffer o kommt die Bezeichnung SCHWARZ

Ziffer	Bezeichnung
o	SCHWARZ
1	BRAUN
2	ROT
3	ORANGE
4	GELB
5	GRÜN/OHNE Toleranzring
6	BLAU
7	VIOLETT
8	GRAU/SILBER
9	WEISS/GOLD

Listing:

```
o01  LBL "OHM"     Berechnen des Widerstandes
o02  1 E 2
o03  /
o04  INT
o05  STO o1
o06  LAST X
o07  FRC
o08  1o
o09  *
o1o  STO oo
o11  INT
o12  STO o2
o13  8
o14  X>Y?
o15  GTO o1
o16  -1
o17  STO o2
o18  RCL Z
o19  9
o2o  X=Y?
o21  GTO o1
o22  -2
o23  STO o2
o24  LBL o1
o25  RCL o2
o26  1o^x
o27  ST  o1
o28  RCL oo        Berechnen der Toleranz
o29  FRC
o3o  1o
o31  *
o32  INT
o33  STO o2
o34  3
o35  X>Y?
o36  GTO o2
o37  1o
o38  STO o2
o39  RCL Z
o4o  8
o41  X=Y?
o42  GTO o2
o43  5
o44  STO o2
o45  RCL Z
o46  9
o47  X=Y?
o48  GTO o2
o49  2o
o5o  STO o2
o51  LBL o2        Anzeige-Routine
o52  FIX o
o53  CLA
o54  ARCL o2
o55  ├"%  "
o56  ENG 2
o57  ARCL o1
o58  AVIEW
o59  END
```

Name: DIN-ASA - Umrechnung

Beschreibung: Das Programm rechnet DIN-Werte in ASA-Werte um. Es ist für alle Fotofreunde interessant, die keine Umrechnungstabelle besitzen oder auf deren Apparate keine 2 Skalen aufgedruckt sind. Die DIN- und ASA-Werte zeigen die Empfindlichkeit eines Films an. Je höher der Wert, um so lichtempfindlicher ist der Film.

Entwicklung: Raimund Berg, Hagener Str. 2oo, 591o Kreuztal 6

Register: 22

Speicher: STO oo Eingabewert

Status: beliebig, ᵀAD und ᵀDA können irgendwelchen Tasten zugeordnet werden.

Ausführung: 1. ASA-Wert eingeben ($3 \le x \le 128oo$) , XEQ ᵀAD
2. DIN-Wert eingeben ($6 \le x \le 42$) , XEQ ᵀDA

Listing:

```
oo1  LBL ᵀDA       DIN → ASA
     FIX o         Status festlegen
     CF 29
     INT           etwaiger Dezimalteil
     STO oo        unterdrücken, abspeichern
     3
     MOD
     GTO IND X     die passende Reihe aussuchen
oo9  LBL oo        1. Reihe
     128oo         Anfangswert
     XEQ o3
     INT
o13  LBL o4        Anzeige-Routine
     CLA
     ARCL oo
     ⊢' DIN= '
     ARCL X
     ⊢' ASA'
     GTO o6
o19  LBL o2        2. Reihe
     1 E 4         Anfangswert
     XEQ o3
     7oo
o23  LBL o5        Routine, die die Unregelmäßig-
     x<> y         keit der Reihen ausgleicht
     x>y?
     GTO o4
     1,o24
     *
     GTO o4
o31  LBL o1        3. Reihe
     8 E 3         Anfangswert
     XEQ o3
     7o
     GTO o5
o36  LBL o3        ASA-Wert berechnen
     2             Anfangswert / 2^(INT|DIN/3 - 14|)
     RCL oo
     3
     /
     14
     -
     ABS
     INT
     y^x
     /
     RTN
o48  LBL ᵀAD       ASA → DIN
     FIX o         Status festlegen
     CF 29
     STO oo
     1 E 2         Umrechnung
     /             3 · ln(ASA/1oo) / ln 2 + 21
     LN
     2
     LN
     /
     3
     *
     21
     +
     RND
     CLA
     ARCL oo
     ⊢' ASA= '
     ARCL X
     ⊢ DIN'
o68  LBL o6        Standardmodus her-
     AVIEW         stellen
     SF 29
     FIX 2
o72  END
```

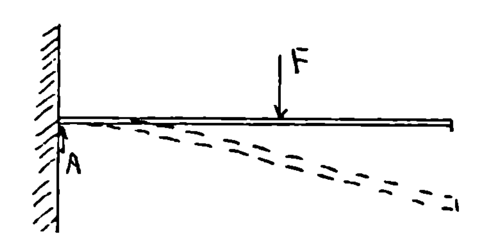
ASA-Wert: $\frac{\text{Anfangswert}}{2^{\,INT\left|\frac{DIN}{3} - 14\right|}}$

Umrechnung: $3\frac{\ln \frac{ASA}{1oo}}{\ln 2} + 21$

Titel des Programms:	Formänderung am Kragarm
Name des Programmierers:	Hans-Günter Lütke Uphues
Rechnermarke und Typ:	HP-41C
Softwaremodul:	keines
Anzahl der Speicher/Pgmzeilen	16/413
Drucker:	ja und nein (wahlweise)
Anzahl der Kartenseiten:	6

PROGRAMMBESCHREIBUNG / PROBLEMSTELLUNG:

Die Beanspruchung eines eingespannten Trägers, hervorgerufen durch ein Biegemoment, hat eine Krümmung der Trägerachse zur Folge:



A: Gegenkraft
bei A: Einspannstelle

Die deformierte Trägerachse nennt man Biegelinie oder auch elastische Linie.

Dieses Pgm ist so aufgebaut, daß verschiedene Belastungsfälle überlagert werden, d.h. daß die einzelnen Belastungsfälle einzeln nacheinander durchgerechnet werden (siehe Pgm-Test) und alle Zwischenergebnisse angezeigt x oder beim angeschlossenen Drucker mit Text ausgedruckt werden. Auf Wunsch kann auch jederzeit die Ausgabe der Summenregister erfolgen.

Registerbelegung:

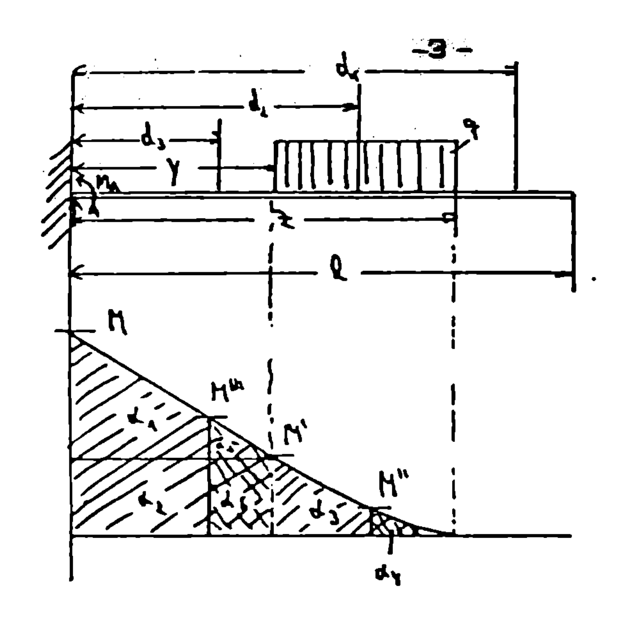
00: EI	01: d	02: y	03: z
04: q	05: x	06: F	07: Summe A
08: Summe M	09: Summe phiA	10: Summe phid	11: Summe f

12,13, 14 und 15 benutzt

F = Kraft (N)   M = Moment (N/mm)
f = Durchbiegung (mm)   E = Elastizitätsmodul (N/mm^2)
I = Trägheitsmoment (mm^4)   phi = Winkel (rad)
q = Streckenlast (N/mm)   d, y, z und x = Längen (mm)

PROGRAMMBEDIENUNG:

Bedienungsanweisung:	Eingabe	Taste	Anzeige + Ausgabe	Berechnung
1. Eingabe der Systemwerte		XEQ KRAG	F?	
	E	R/S	I?	
	I	R/S	d?	
	d	R/S	d?=....	
2. Eingabe und Berechnung				
		A	X Y?	Auflager A
	y	R/S	Z?	Moment M_A
	z	R/S	X Q?	Drehwinkel A
	q	R/S		Drehwinkel d
				Durchbiegung
		B	X?	
	x	R/S	F?	
	F	R/S		"
3. Ausgabe Summenspeicher:		E	Summe A	
		R/S+	Summe M	
+): Wenn Drucker angeschlossen: R/S nicht notwendig!		R/S+	Summe phi	A
		R/S+	Summe phi	d
		R/S+	Summe f	



benützte FORMELN:

Kragarm mit Streckenlast:

$A = q(z - y)$

$M = q\left(\frac{z^2}{2} - \frac{y^2}{2}\right)$

$M' = \frac{q(z - y)^2}{2}$

$z = b$

$y = a$

$$f = \frac{M - M'}{EI}\,\frac{a}{2}\left(d - \frac{a}{3}\right) + \frac{M'a}{EI}\left(d - \frac{a}{2}\right) + \frac{M'(b-a)}{3EI}\left(d - \frac{b + 3a}{4}\right)$$

Fall 1: $d \geq b$

$f_{d1} = f$

$phi_{d1} = phi_A$

Fall 2: $a \leq d < b$

$M'' = \left(\frac{b-d}{b-a}\right)^2 M'$

$f_{d2} = f + \frac{M''(b-d)^2}{12EI}$

$phi_{d2} = phi_A - \frac{M''(b-d)}{3EI}$

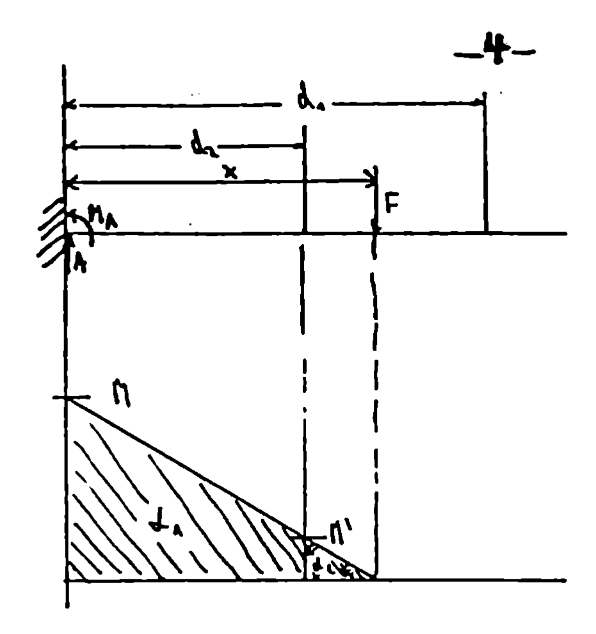
Fall 3: $d < a$

$M''' = M\left(1 - \frac{d}{a}\right) + \frac{M'd}{a}$

$f_{d3} = f - \frac{M'(b-a)}{3EI}\left(d - \frac{b+3a}{4}\right) + \frac{(M''' + 2M')(a-d)^2}{6EI}$

$phi_{d3} = phi_A - \frac{M'(b-a)}{3EI} - \frac{(M''' + M')(a-d)}{2EI}$

$$phi_A = \frac{M - M'}{2EI}a + \frac{M'a}{EI} + \frac{M'(b-a)}{3EI}$$



benutzte FORMELN:

Kragarm mit Einzellast:

$$M = Fx$$

$$f = \frac{Mx}{2EI}\left(d - \frac{x}{3}\right)$$

$$M' = M\left(1 - \frac{d}{x}\right)$$

$$phi_A = \frac{Fx^2}{2EI}$$

Fall 1: $d \geq x$

$f_{d1} = f$
$phi_{d1} = phi_A$

Fall 2: $d < x$

$f_{d2} = f + \frac{M'(x-d)^2}{6EI}$
$phi_{d2} = phi_A - \frac{M'(x-d)}{2EI}$

PGM . TEST

```
E?=210,000.0000
I?=10,000,000.00
d?=2,150.00000

y?=150.00000
z?=3,000.00000
α?=11.00000

A=31,350.00000
M=49,376,250.00
∡A=0.02357
∡d=0.02303
f=33.10771

ΣA=31,350.00000
ΣM=49,376,250.00
∡A=0.02357
∡d=0.02303
Σf=33.10771

x?=2,150.00000
F?=100,000.0000

A=100,000.0000
M=215,000,000.0
∡A=0.11006
∡d=0.11006
f=157.75198

ΣA=131,350.0000
ΣM=264,376,250.0
∡A=0.13363
∡d=0.13309
Σf=190.85969
```

```
E?=210,000.0000
I?=10,000,000.00
d?=100.00000

y?=150.00000
z?=3,000.00000
α?=11.00000

A=31,350.00000
M=49,376,250.00
∡A=0.02357
∡d=0.00228
f=0.11507

ΣA=31,350.00000
ΣM=49,376,250.00
∡A=0.02357
∡d=0.00228
Σf=0.11507

x?=2,150.00000
F?=100,000.0000

A=100,000.0000
M=215,000,000.0
∡A=0.11006
∡d=0.01000
f=0.50397

ΣA=131,350.0000
ΣM=264,376,250.0
∡A=0.13363
∡d=0.01228
Σf=0.61904
```

```
E?=210,000.0000
I?=10,000,000.00
d?=3,500.00000

y?=150.00000
z?=3,000.00000
α?=11.00000

A=31,350.00000
M=49,376,250.00
∡A=0.02357
∡d=0.02357
f=64.81123

ΣA=31,350.00000
ΣM=49,376,250.00
∡A=0.02357
∡d=0.02357
Σf=64.81123

x?=2,150.00000
F?=100,000.0000

A=100,000.0000
M=215,000,000.0
∡A=0.11006
∡d=0.11006
f=306.33234

ΣA=131,350.0000
ΣM=264,376,250.0
∡A=0.13363
∡d=0.13363
Σf=371.14357
```

```
01◆LBL "KRAG"
02 FIX 5
03 CLRG
04 CF 13
05 "E?"
06 PROMPT
07 "⊢="
08 ARCL X
09 AVIEW
10 "I?"
11 PROMPT
12 "⊢="
13 ARCL X
14 AVIEW
15 *
16 STO 00
17 SF 13
18 "d?"
19 PROMPT
20 "⊢="
21 ARCL X
22 AVIEW
23 STO 01
24 ADV
25 STOP
26◆LBL B
27 SF 13
28 "X?"
29 PROMPT
30 "⊢="
31 ARCL X
32 AVIEW
33 STO 05
34 CF 13
35 "F?"
36 PROMPT
37 "⊢="
38 ARCL X
39 AVIEW
40 STO 06
41 ADV
42 "A="
43 RCL 06
44 ST+ 07
45 ARCL X
46 AVIEW
47 FC? 55
48 STOP
49 "M="
50 RCL 05
51 *
52 ST+ 08
53 ARCL X
54 AVIEW
55 FC? 55
56 STOP
57 "∡A="
58 RCL 05
59 *
60 2
61 /
62 RCL 00
63 /
64 ST+ 09
65 ARCL X
66 AVIEW
67 STO 12
68 RCL 01
69 RCL 05
70 3
71 /
72 -
73 *
74 STO 13
75 FC? 55
76 STOP
77 RCL 05
78 RCL 01
79 X<Y?
80 GTO 02
81 RCL 12
82 "∡d="
83 ST+ 10
84 ARCL X
85 AVIEW
86 FC? 55
87 STOP
88 RCL 13
89 GTO 16
90◆LBL 02
91 RCL 06
92 RCL 05
93 *
94 1
95 RCL 01
96 RCL 05
97 /
98 -
99 *
100 2
101 /
102 RCL 00
103 /
104 RCL 05
105 RCL 01
106 -
107 *
108 STO 14
109 LASTX
110 *
111 3
112 /
113 ST+ 13
114 RCL 14
115 CHS
116 RCL 12
117 +
118 ST+ 10
119 "∡d="
120 ARCL X
121 AVIEW
122 FC? 55
123 STOP
124 RCL 13
125 GTO 16
126◆LBL A
127 SF 13
128 "Y?"
129 PROMPT
130 "⊢="
131 ARCL X
132 AVIEW
133 STO 02
134 "Z?"
135 PROMPT
136 "⊢="
137 ARCL X
138 AVIEW
139 STO 03
140 "Q?"
141 PROMPT
142 "⊢="
143 ARCL X
144 AVIEW
145 STO 04
146 ADV
147 CF 13
148 RCL 04
149 RCL 03
150 RCL 02
151 -
152 *
153 ST+ 07
154 "A="
155 ARCL X
156 AVIEW
157 FC? 55
158 STOP
159 LASTX
160 *
161 2
162 /
163 STO 12
164 RCL 03
165 X↑2
166 RCL 02
167 X↑2
168 -
169 RCL 04
170 *
171 2
172 /
173 ST+ 08
174 STO 13
175 "M="
176 ARCL X
177 AVIEW
178 FC? 55
179 STOP
180 RCL 12
181 RCL 03
182 3
183 /
184 RCL 02
185 6
186 /
187 +
188 *
189 RCL 13
190 RCL 02
191 *
192 2
193 /
194 +
195 RCL 00
```

```
196 /
197 STO 14
198 ST+ 09
199 "∠A="
200 ARCL X
201 AVIEW
202 FC? 55
203 STOP
204 RCL 13
205 RCL 12
206 -
207 RCL 02
208 2
209 /
210 *
211 RCL 01
212 RCL 02
213 3
214 /
215 -
216 *
217 RCL 01
218 RCL 02
219 2
220 /
221 -
222 RCL 02
223 *
224 RCL 12
225 *
226 +
227 RCL 02
228 3
229 *
230 RCL 03
231 +
232 4
233 /
234 RCL 01
235 -
236 RCL 02
237 RCL 03
238 -
239 *
240 3
241 /
242 RCL 12
243 *
244 +
245 RCL 00
246 /
247 STO 15
248 RCL 02
249 RCL 01
250 X<Y?
251 GTO 01
252 RCL 03
253 X<=Y?
254 GTO 15
255 -
256 RCL 03
257 RCL 02
258 -
259 /
260 X↑2
261 RCL 12
262 *
263 RCL 00
264 /
265 3
266 /
267 RCL 03
268 RCL 01
269 -
270 *
271 STO 12
272 LASTX
273 *
274 4
275 /
276 ST+ 15
277 RCL 14
278 RCL 12
279 -
280 ST+ 10
281 "∠d="
282 ARCL X
283 AVIEW
284 FC? 55
285 STOP
286 RCL 15
287 GTO 16
288♦LBL 01
289 RCL 13
290 1
291 RCL 01
292 RCL 02
293 /
294 -
295 *
296 RCL 12
297 RCL 01
298 *
299 RCL 02
300 /
301 +
302 STO 13
303 RCL 12
304 +
305 2
306 /
307 RCL 02
308 RCL 01
309 -
310 *
311 RCL 03
312 RCL 02
313 -
314 3
315 /
316 RCL 12
317 *
318 +
319 RCL 00
320 /
321 CHS
322 RCL 14
323 +
324 ST+ 10
325 "∠d="
326 ARCL X
327 AVIEW
328 FC? 55
329 STOP
330 RCL 13
331 RCL 12
332 2
333 *
334 +
335 6
336 /
337 RCL 02
338 RCL 01
339 -
340 X↑2
341 *
342 RCL 03
343 RCL 02
344 3
345 *
346 +
347 4
348 /
349 RCL 01
350 -
351 RCL 02
352 RCL 03
353 -
354 *
355 3
356 /
357 RCL 12
358 *
359 -
360 RCL 00
361 /
362 RCL 15
363 +
364 GTO 16
365♦LBL 15
366 RCL 14
367 ST+ 10
368 "∠d="
369 ARCL X
370 AVIEW
371 FC? 55
372 STOP
373 RCL 15
374♦LBL 16
375 SF 13
376 ST+ 11
377 "F="
378 ARCL X
379 AVIEW
380 CF 13
381 ADV
382 ADV
383 STOP
384♦LBL E
385 "ΣA="
386 ARCL 07
387 AVIEW
388 FC? 55
389 STOP
390 "ΣM="
391 ARCL 08
392 AVIEW
393 FC? 55
394 STOP
395 "∠A="
396 ARCL 09
397 AVIEW
398 FC? 55
399 STOP
400 "∠d="
401 ARCL 10
402 AVIEW
403 FC? 55
404 STOP
405 SF 13
406 "ΣF="
407 ARCL 11
408 AVIEW
409 CF 13
410 ADV
411 ADV
412 ADV
413 ADV
414 END
```

IMPRESSUM — HEWLETT-PACKARD-ANWENDER-CLUB

1) Satzung:

a) Eintritt: Der Eintritt ist jederzeit möglich.

b) Beitrag: Der Jahresbeitrag beträgt z.Zt. 4o,--DM.

c) Leistungen: Zweck des Vereins ist die Förderung von Wissenschaft und Forschung. Der Satzungszweck wird verwirklicht insbesondere durch Durchführung wissenschaftlicher Veranstaltungen und Forschungs- und Entwicklungsarbeiten, sowie der Herausgabe eines vereinseigenen Informationsblattes.
Letzteres erhalten Clubmitglieder kostenlos; ein Verkauf an Nichtmitglieder ist nicht möglich.
Weitergehende Forderungen, auch was Umfang und Erscheinungsdatum der Informationen betrifft, sind nicht möglich.

d) Organisation: Der Club ist völlig frei organisiert und steht in keiner direkten Verbindung mit Hewlett-Packard GmbH oder irgendeiner anderen Organisation.

e) Austritt: Der Austritt ist jederzeit möglich. Da der Beitrag aber jeweils für ein Kalenderjahr entrichtet wird, endet die Mitgliedschaft immer zum 31. Dezember des Jahres, in dem die Kündigung ausgesprochen wurde. Aus organisatorischen Gründen hat der Austritt bis zum 1.Dezember des jeweiligen Jahres zu erfolgen.

f) Copyright: Alle in den Clubheften erschienen Artikel dürfen nur mit der ausdrücklichen Genehmigung des Autors vervielfältigt, in Datenverarbeitungsanlagen gespeichert oder in einer anderen Weise, die über den persönlichen Rahmen hinausläuft, genutzt werden. Dies gilt insbesondere für die Mitgliederlisten, für die eine solche Erlaubnis überhaupt nicht ausgesprochen werden kann.

g) Haftung: Weder der Club als ganzes, noch einzelne Mitglieder oder dritte Personen oder Einrichtungen können in irgendeiner Weise für Dinge oder Handlungen, insbesondere für Programme und andere Veröffentlichungen, sowie den Schaden der durch selbige angerichtet werden könnte, haftbar gemacht werden.
Für namentlich gekennzeichnete Berichte und andere Publikationen trägt allein der Autor die Verantwortung.

) Ergänzungen und Erläuterungen:

zu a) Eintritt: Beim Eintritt wird eine Aufnahmegebühr erhoben.

zu b) Beitrag: Beitragserhöhungen müssen bis zum 15.November eines jeden Jahres jeweils für das Folgende Jahr bekanntgegeben werden. Ebenso Beitragsermäßigungen, nennenswerte Änderungen der Clubleistungen und Auflösung des Clubs. Diese Dinge sind zuvor auf einer Mitgliederversammlung zu besprechen.

zu c) Leistungen: 1981 sind acht Ausgaben vorgesehen.

zu d) Organisation: derzeitige Struktur s.u.

zu g) Haftung: Die Mitglieder erhalten die Publikationen auf eigenes Risiko hin und haben sie daher selbst zu untersuchen. Weder der Autor noch eine sonstige Per-

son oder Organisation übernehmen für das Programm- und sonstige Material eine irgendwie geartete Gewährleistung oder Haftung. Dies gilt auch für indirekte oder Folgeschäden.
Die Ergänzung "Für namentlich gekennz.Berichte u.a. Publ.trägt allein der Autor die Verantw." wurde hinzugefügt, um zu gewährleisten, daß niemand Schriftstücke einsendet, die bereits anderweitig mit einem Copyright versehen sind.

Weitere Erläuterungen und Ergänzungen:

a) Die Aktivitäten des Clubs galten bisher dem HP-41c. Sie werden es auch auf absehbare Zeit.

b) Die Beiträge und Aufnahmegebühren werden verwendet für:
- Druck der Publikationen - Portokosten
- Druck von Werbeschriften, Briefbögen u.ä.
- Kauf von Papier,Umschlägen,Stempeln u.s.w.
- Kauf von Fachliteratur für die Clubbibliothek
- Kauf von Computer-Hardware und Software (insbesondere auch bei Erweiterung des Tätigkeitsfeldes über den HP-41c hinaus)
- Aufwandsentschädigungen für besonders aktive Mitglieder oder Nichtmitglieder für organisatorische Arbeiten, wie z.B. Versand- und Sortierarbeiten, Übersetzen umfangreicher Barcodelisten, Zusammenstellung der Clubinfos, Aufnahme neuer Mitglieder u.ä.
- Durchführung wissenschaftlicher Veranstaltungen, die im Interessengebiet der Clubmitglieder liegen.

---

3) Aktive Mitglieder und Aktivitätsgebiete (Mitgl.Nr. —— Name&Adr. : siehe Mitgl. Liste)

Mitgl. Nr.	Zuständig für/ Leiter von/ Organisation von/
1	Aufnahme neuer Mitglieder : Austritt : Finanzen : Zusammenstellung der Infos und übergeordnete Organisation : Versand der Clubinfos : Planung neuer Projekte :
2	Druck der Clubinfos : Plotten der Barcodes
3	Sammelbestellungen (in finanzieller und rechtl. Sicht unabhängig vom Club); Projektleitung "intelligente Spiele" :
11	Projektleitung und Autor "Programmierhilfen für Anfänger"
25	Mitgliederliste
75	Projektleitung und Autor "Synthetische Prgr."
81	Projektleitung "Kalenderrechnungen"
116	Projektleitung "Magnetkartenbibliothek" (s.a.205)
117	Projektleitung und Autor "Bücherecke" : Verwalter der Clubbibliothek: Interfacetechniken
164	Sammelbestellungen Akkus,Zubehör u.ä. (Anmerk. wie (3))
205	Projektleitung "Magnetkartenbibliothek" (s.a.116)

Gesucht: Projektleiter für "Statik","Mathematik","Statistik", "Elektrotechnik","Plottprogramme","Spiele" u.v.a.
WER MÖCHTE ?????

Stand der Informationen dieser Seite: Ostern 1981

Achtung: Die Seitennummern 39-81 bis 42-81 wurden versehentlich doppelt vergeben ! Die reinen BarCode-Seiten bitte mit [illegible] bezeichnen !

Mitgliederliste

Anmerkungen: Diese Liste wird ständig aktualisiert. Änderungen und Ergänzungen können mir jederzeit übermittelt werden. Bis Mitgliedsnummer 27 ist die Nr. mehr oder weniger zufällig, danach abhängig vom Zeitpunkt des Clubeintritts. Oliver

---

1 Oliver Rietschel (63)
Bismarckstr. 31
2420 Eutin
Tel. 04521/2813

Schüler; Mathematik und Chemie; Assistent in Informatik
Kartenleser, Drucker, Strichcodeleser, Mathe ROM, Memorys
Pascal, Basic(NorthStar, Wang, Cromemco), Assembler, Maschinensprache (Z8080A, Z80), Algol 60,
H: Rudern, Kegeln, Schwimmen, Musik(HiFi), Radfahren, Reisen

2 Detlev Bock (57)
Petrikirchstr. 36
3400 Göttingen
Tel. 0551/33221

Student; Physik und Elektrotechnik
Kartenleser, Drucker, Strichcodeleser, ROM's, DIM's
Fortran IV, Algol, Basic, } HP-85
H: Lesen

3 Ulrich Davertzhofen (57)
Ostenbergstr. 97/ Z.8
4600 Dortmund 50

Student; Statistik
Memorys
Basic
H: Spielen; Labyrinthe (Bücher, Modelle, & Hecken Labyr. Besuch von Rasen)

4 Gerald Kirsch (56)
Mühlweg 8
6922 Meckesheim
Tel. 06226/8880

Student; Nachrichtentechnik
Kartenleser, Memory
Basic
H: Elektronik, Tischtennis

5 Raimund Berg (57)
Hagener Str. 200
5910 Kreuztal 6
Tel. 02732/80240

Maler-und Lackierer
Kartenleser, Memorys
-
H: Gitarrespielen, Fotografieren, betriebswirtsch. Probleme

6 Rudolf Eber (37)
Wetzldorfer Str. 12
8481 Erbendorf
Tel. 09682/2379

Verwaltungs-Angestellter
Memorys
-
H: Amateurfunk (DL8KE), Elektronik

7 Erich Graller (47)
Hofackerweg 19
8752 Goldbach 2
Tel. 06021/53544

Technischer Berater für Bekleidungsindustrie
Kartenleser, Memorys
AOS
H: Gitarre, Bergwandern

8 Siegfried Gruhler (53)
Sigmarswangenstr. 3
7243 Vohringen 2
Tel. 07454/2764 priv.
07423/70227 Bü

Ing. grad. Feinwerktechnik
Kartenleser, Drucker, Standardmodul, Memorys
Fortran IV
H: Musik(Saxophon, Akkordeon), "Häuslebauen", Fußball

9 Gert-Jürgen Grünler (47)
Düsseldorfer Str. 56a
1000 Berlin 15
Tel. 030/8817440

Studienrat; Mathematik, Physik
Kartenleser, Drucker, Strichcodeleser, ROM's, Memorys
Basic
H: Fotografieren, Sport, Musik

10 Friedrich Hillebrandt (48)
Jülicher Str. 443
5100 Aachen

Postarbeiter
Kartenleser, Drucker, Mathe ROM, Memorys
Fortran IV, Algol 60, Basic
H: Deutsche Literatur des frühen 20. Jh., Krimis
Besitzer von: 41,25,19,65 (HP's) und 57,59 (TI's)
Casio FX-502P

11 Klaus Werner Hoenow (38)
Alaskaweg 18
2000 Hamburg 73
Tel. 040/6786959 priv.
040/2719499 Bü

Diplom-Physiker
Kartenleser, Drucker, ROM's, Memorys
Fortran 77, Fortran IV, Basic, Algol 60
H: Technikgeschichte, Astronomie, Schach, Photographie

12 Andreas Krenn (64).
Elisabethstr. 35
A-25oo Baden
Tel. o2252/82262

Schüler; HTL Nachrichtentechnik
Magnetkartenleser, Drucker, Statistik ROM, Memorys
-
H:Mathematik, Science Fiction

13 Harald M. Krumins (55)
Doroteenstr. 62
233o Eckernförde
Tel. o4351/84935

72-75:Industriekaufm.,75-78:Wirtschaftsgymn.,z.Zt. Soldat
- ab1981: Studium Wirtsch.-Pädagogik
Basic
H: Audio, SF-Literatur, SF-Filme, Hard-Software

14 Günter Lelarge
Kelterbaum 13
547o Andernach 14
Tel. o2636/2137

16 Clemens Mirgel (43)
Weserstr. 7b
6o74 Rödermark
Tel. o6o74/7813

Realschullehrer; Mathematik,Physik,Chemie
Kartenleser; Memorys
-
H:Briefmarken (Deutschl. & Frkr.), Heimwerken
Fotografieren/Filmen, Entwickeln(auch Farbe)

17 Sidgmar Ortner
Lahrndorfer Str. 131
A-4451 Garsten

18 Georg Raabe (48)
Brotweg 1o
332o Salzgitter 31
Tel. o5341/265848

Dipl. Ing. ; Bauwesen
Kartenleser, Drucker, Memorys
Fortran IV, Basic, Algol 6o, APL, pdp/8-Focal, HPL
H:Film, Foto, Astronomie, Briefmarken
Besitze einen HP-67 mit Mathe und Statik Programmen

19 Raoul Marc Ramdohr (58)
Markstr. 19o
463o Bochum
Tel. o234/73614 Zimmer 22

Student; Chemie/Diplom
Kartenleser, Memorys
Fortran, Algol, Basic, AOS
H: Fotografieren,SW-Entwicklung und Abzüge, Tennis
Ski, Tauchen, Fahrradtouren, Schlafen

2o Heinz-Dieter Schmitt
Cassdorfer Weg 5
3588 Homberg/Efze

21 Dieter Schuppan (3o)
Schiller Str. 1o
55o3 Konz
Tel. o65o1/3131

Apotheker
Kartenleser, Drucker, Module

22 Manfred Sures
Eschenweg 4
7519 Eppingen-Rohrbach
Tel. o7262/1386

23 Andreas Weiler (65)
Wienerstr. 64
A-25oo Baden/Wien
-

Schüler; Nachrichtentechnik
Kartenleser, Memory
Pascal, Basic, Maschinensprache
H: Minigolf, Astronomie, Elektronik, Mathematik, HiFi

24 Jürgen Weise
Bergfriedstr. 1oa
1ooo Berlin 61

25 Edmund Weitz (65) Heinrich-Heine-Str. 1o1 3151 Essinghausen Tel. o5171/17718	Schüler; Mathematik,Politik Magnetkartenleser, Memory Basic,PL/1,Fortran,Algol H: Jazz,Personal-Computer (Apple)
26 Michael Welter Mozartsr. 2 6682 Ottweiler	
27 Manfred Weyand Sauerbruchstr. 25 413o Moers 2 Tel. o2841/63931	Bauingenieur
28 Heinrich Henze (24) Zum Hohen Brunnen 12a 34oo Göttingen-Herberhausen 1 Tel. o551/397717 Bü	Rundfunkmechaniker-Meister; Elektronik(Werkstattleiter) Kartenleser, Memorys H: Fotografieren, basteln in Metall u. Holz für Haus und Garten
29 Emil Wörndle Talheim 58 I-39o5o Völs -	Student; Architektur Memory - H: Modellbau,Motorrad,klass.Musik,Bousai+Garten
3o Christof Born (63) Gartenstr. 17 466o Gelsenkirchen Tel. o2o9/771878	Schüler; Mathematik, Physik Kartenleser, Mathe-ROM Basic, Pascal H: el. Keyboards Arbeite in der Schule an: WANG 22oo C
31 Dieter Peppmüller (51) Nikolausstr. 4o 5173 Aldenhoven-Schleiden Tel. o2464/8547 pr. o24o2/23o75 Bü	Bankkaufmann Kartenleser, Drucker, Module - H: z.Zt. Hausbau -- Heizungstechnik
32 Wolfgang Treutwein (33) Vorjurastr. 124 85oo Nürnberg 6o Tel. o911/63575o	Bauingenieur (grad.) , Ingenieurbau Kartenleser, Module, Netzanschl., HP-35/25/67 H: Bücher, klass.Musik, Fotografie, Pfeifenrauchen
33 Karl-Heinz Spyra (49) Sternstr. 11 58oo Hagen 1 Tel. o2331/331927	Elektriker Kartenleser, Memorys H: Schach mit Computer, Basteln mit Elektronik
34 Erwin Hartmann (36) Berliner Str. 56 4o3o Ratingen Tel. o21o2/4526o	EDV-Ingenieur Memory Basic, Assembler H: Foto,Film, Abenteuer-Reisen
35 Dr. Hans La Roche Wildunger Str. 52 7ooo Stuttgart-Bad Cannstadt Tel. o711/567656	Apotheker Kartenleser,Drucker § Basic, HP Language 9815/S H:
36 Axel Burkart (6o) Camphausenstr. 4o 43oo Essen 1 Tel. o2o1/7o6892	Student(Dortmund), HF: Informatik NF: Elektrotechnik Memorys PDP11 Assembler, Pascal, SC/MP, Simula, 68o9 Maschin. H: Mikroprozessoren, Lesen, Musik(Anfänger)
37 Falko-Romeo Herda (27) Ortlerweg 44 1ooo Berlin 45 Tel. o3o/8176388	Verwaltung, Kartenleser, Drucker, Memorys AOS H: Yoga, Gitarre

38 Udo Ricken (57) ~~Akademiestr. 46~~ Pfinztalstr. 17 7500 Karlsruhe 41 Tel. --	[illegible]ent; Wirtschaftsingenieur [illegible]tenleser, Memory - H: Radfahren, Schwimmen, Tennis
39 Helmut Gärtner Postfach 700 760 6000 Frankfurt/M. 70	
40 Dr. Hans-Wilhelm Helberg (28) Plauener Str. 1 3400 Göttingen Tel. --	Physiker Memorys Fortran
41 Siegfried Schwarze (65) Hilblestr. 34 8000 München 19 Tel. 089/193649	Schüler; Humanist. Gymnasium Memorys (Pascal),(Basic) H: Motorräder, Lesen, Chemie
42 Ralf Wanser (61) Am Erdbeerstein 31 6240 Königstein Tel. 06174/5587	Schüler; Physik, Mathematik Kartenleser, Memory - H: Segelfliegen, Skifahren
43 Egon Irriger Alexanderstr. 85 4223 Voerde 1	
44 Norbert Weber (60) Val Ste Croix 147 Luxembourg Tel. 444747	Student; Wissenschaften (Bio, Chemie, Physik, Mathe) Kartenleser, Drucker, Memorys (DDM), Mathe ROM&Stand. Basic, AOS H: Computer, Mineralien, Waffen, Biochemie
45 Klaus-Peter Nuber (50) Briennerstr. 42 8000 München 2 Tel. 089/ 533066 oder 539376 Bü 089/ 522501 pr.	Vermessungsingenieur (grad.) Kartenleser, Drucker, Memorys
46 Helmut Maurer Alleestr. 21 CH-8280 Kreuzlingen Tel. 072/723706	Ingenieur; Maschinenbau; Schaltwerks& Regelbetr. Entw.& Konstr Magnetkartenleser, Drucker, Moduln - H: Segeln (binnen + See)
47 Heinz Elsässer (48) Veilchenstr. 72 7537 Remchingen Tel. 07232/71571	Selbständig; Ingenieur; Nachrichten-Daten-Technik Kartenleser, Drucker, Memorys Pascal, Basic, Cobol H: Foto, Windsurfing
48 Ludwig Held (22) Löwentorstr. 65 7000 Stuttgart 50 Tel. 0711/8106237	Leiter Betriebsorganisation und Datenverarbeitung Kartenleser, Drucker, Memorys Basic(IBM5120),Cobol(ANS),Fortran,PL/1-Optimizer, APL, Autocoder, SPS, AOS
49 Hans Gaudig (43) Marktstr. 15 7230 Schramberg Tel. 07422/4289	Apotheker Kartenleser, Drucker - H: Modellflug
50 Christof Waas Gleinkergasse 16 A-4400 Steyr Tel. 07252/639323	Luftfahrzeugmechaniker, ab Sept.80: Schüler; HTL- Kartenleser,Memory,Akku Pac /// Nachrichtentechnik - H: Elektronik(Anfänger), Mikrocomputer, Motorräder, Foto;SW: kleines SW-Fotolabor vorhanden

51 Michel Majerus (63) Heiderscheid Rue de Eglise 18 Luxembourg Tel. oo352/89o22	Schüler; Naturwissenschaften Kartenleser,Memorys Basic H: Motorräder,Schwimmen,Lesen,HardRock
52 Gerhard Siewert (14) Mozartsr. 14 645o Hanau 1 Tel. o6181/82464	Ingenieur; Feinwerktechnik, Elektronik
53 Michael Tarnowski (61) Fritz-Kalle-Str. 4 62oo Wiesbaden Tel. o6121/844118	Schüler; Mathematik, Physik Kartenleser, Drucker, Memorys Basic H: Bücher, Naturwissenschaftl. Themen, hybonet. Probl.
54 Dr. Johannes Ahlert (49) Heinrich-Ohm-Str. 24 3474 Boffzen Tel. o551/39765o Bü o5271/5248 pr.	Dipl.Physiker; Kernphysik Kartenleser, Drucker Fortran, Algol H: Squash, Bergsteigen, vieles mehr
55 Uwe Consbruch Wallstr. 9 4o3o Ratingen 1	
56 Walter Kropf (29) Schönowitzstr. 14 A-87oo Leoben Tel. o3862/2571	Betriebsingenieur; Elektrotechnik Kartenleser, Drucker, Memorys - H: Elektronik, Digitaltechnik, Bergsteigen, Wandern
57 Hans Mann Mühlenstr. 5 34o1 Ebergötzen	
58 Wilhelm Theisen (58) Egerstr. 36 46oo Dortmund 5o Tel. o231/7187o8	Student; Elektrische Energietechnik - Fortran H: Handball, Aquaristik
59 Guido R. Studer (62) Turnhallenweg 7a CH-4143 Dornach Tel. o61/724978	kaufm. Lehrling (2.Jahr), kaufm.Schule+EDV Memorys Basic(68oo),HP-Basic(HP-25o)(HP-85),(Cobol),(Pascal) H: Musik(HiFi), Modellbau, Schreiben, Sammeln, Lesen u.v.m.
6o Robert Klauc (63) Stiegasse 9/35/9 A-11oo Wien Tel. --	Schüler, Elektrotechnik Kartenleser,Drucker,Memorys, Mathe ROM Basic, Focal H: Modellbau
1 Hagen Burkardt (5o) Sauerstr. 51 623o Frankfurt-Nied. 8o Tel. o611/397834	Einzelhandelsfachwirt IHK Kartenleser,Drucker,Memorys Basic H: HiFi, Fotografie, Schmalfilm, Astrologie
2 Stephan Grabiak Schloß Senden 44o3 Senden Tel. o2597/8219	
3 Ulrich Merz (57) Alt-Zeilsheim 6o 623o Frankfurt 8o Tel. o611/362728	Ausbilder; Fernmeldetechnik Kartenleser, Memorys Assembler, Basic, (Fortran) H: Tonübertragungen, Foto, Auto, Elektronik

64 Alexander Zerawa (53) Münchner Bundesstr. 1o A-5o2o Salzburg Tel. o6222/358185	Kfm. Angestellter ; Einkauf - - H: Computer, Wandern/Natur, Familie
65 Reinhold Wurster Friedrichstr. 3 7433 Dettingen Tel. o7123/714o2	
66 Dr. Werner Ehm Postfach 3o48 c/o GAD 44oo Münster	
67 W. Leipold (49) Kopernikusstr. 6b 859o Marktredwitz Tel. o9231/82236	Dipl. Ing. ; Keramik - Anlagenbau - Basic H: Fotografieren, Schmalfilm, Modellflug
68 Gerhard Goder Hüfferstr. 75 // c/o Inst. f. med. Inf. u. Biomath. 44oo Münster Tel. o251 / 83 / 5266 Bü o251/8614o3 pr.	Physiker Kartenleser, Drucker, Strichcodeleser Fortran, PL1, Pascal Interesse: Mathe, Statistik, Synthetische Codes auf dem 41C
69 Andreas Marktscheffel (61) Nachtigallenweg 8 6246 Glashütten 1 Tel. o6174/62oo3	Schüler; Medizin Kartenleser, Memorys Basic, Maschinensprache (Z8o8o) H: Tennis, Tanzen
7o Bernd Malsch (41) Badenerstr. 16a 75oo Karlsruhe-Durlach Tel. o721/4o1292	Apotheker Kartenleser, Drucker - H: Fotografieren, Schach
71 Thomas Fütterer Sonnenhalde 27 774o Triberg Tel. o7722/4749	
72 Alfred Fahr (49) Stettener Str. 25 7758 Meersburg Tel. o7532/6519	Biophysiker Kartenleser, Drucker, Memorys, Statist.&Math-ROM Fortran,Algol(6o,68),Basic,TAS///Thermal&Transp." H: Barockmusik
73 Hagen Klemp (58) Dianastr. 25 1ooo Berlin 28	Ing. (grad.) ; Nachrichtentechnik H: Elektrotechnik, Spiele u.a.
74 Karl Pross Josef Beiser Str. 23 8ooo München 83 Tel. o89/67o1145	
75 Rolf Mach (63) Thorwaldsenstr. 3 6o9o Rüsselsheim Tel. o6142/55485	Schüler ; Mathe, Chemie Kartenleser, Drucker, Memorys Basic, Myp.68oo-Mnemonics H: HiFi, Schallplatten
76 Matthias Grabiak (59) Alexander-Heß-Str. 11 637o Oberursel Tel. o6171/35o8	Student; Physik Kartenleser, Drucker, Mathe ROM, Standard ROM Basic H: Schach /+/ HP-25 Besitz. Zugang: HP-67+PET

77 Günter Tost (51) Adlerstr. 27 668o Neunkirchen Tel. -	Maschinenbau-Techniker Kartenleser, Drucker Basic, Pascal, Miniopt, EXAPT H: Volleyball, Hobby-Fußball, Bücher (techn.)
78 Klaus Jürgen Ries (51) Friedländer Weg 57 34oo Göttingen Tel. o551/59711	Student; Geophysik (Gelernter Großhandelskaufmann) Kartenleser, Drucker, Strichcodeleser, Memorys H: u-Computer, SF-Literatur, Reisen Fortran, Basic, Algol
79 Jörg Warmuth Wartburgstr. 17 1ooo Berlin 62 Tel.	
8o Hermann Winter (46) Hebbelweg 17 588o Lüdenscheid Tel. o2351/53o82	Werkzeugmacher Kartenleser, Memorys H: Elektronik
81 Niels Nöhren (44) Kielort 16 2ooo Norderstedt Tel. o4o/5242888	Postbeamter Kartenleser, Memorys, Finanzrom, Netzanschluß - H: Schach
82 Werner Stroinski (26) Kampweg 7a 1ooo Berlin 27 Tel. o3o/4339813	Hochschullehrer; Elektr. Meßtechnik Kartenleser, opt. Lesestift, Drucker, Memorys - H: Modelleisenbahn, Fotografie
84 Carsten Otto (63) Bismarckstr. 61 2ooo Hamburg 19 Tel. o4o/4914895	Schüler; (Bismarckgymnasium Eimsbüttel) - Basic H: Fotografieren, Basketball, Mathematik, Football
85 Thomas Antweiler (52) Postfach 41o113 5ooo Köln 41 Tel. o221/4o97o7	Student; Bauingenieurswesen Kartenleser, Drucker, Memorys Fortran H: Fotografie, Elektronik
86 Klaus Schmitt (56) Gleiwitzer Str. 44 6944 Hemsbach 2 Tel. o62o1/755o2	Student; Allgem. Elektrotechnik (TH Darmstadt) Kartenleser, Memorys, Mathe-ROM Basic, Pascal, Fortran H: Fotografie, Elektronik, num. Mathematik
87 Torsten Weckmann Postfach 1o2671 463o Bochum 1	
88 Andreas Trögel (64) Grunewaldstr. 13 5o9o Leverkusen 1 Tel. o214/52639	Schüler; Mathematik, Chemie Kartenleser, Memory Pascal H: Mikroprozessoren, Elektronik
89 Rudolf Kern Ludwigstr. 38 7o8o Aalen	
9o Wolfgang Butz (4o) Schinnrainstr. 3a 75oo Karlsruhe 41	Chemiearbeiter Kartenleser, Drucker, Memorys, Akku, Netzteil - H: Foto-Dias, Tonband 1/2 Spur

91 Georg Wehde (2o)
Gartenstr. 17
35o1 Niestetal
Tel. o561/522185

Fachlehrer
Kartenleser,Drucker,Memorys
-
H: Fotografie, Bergwandern

92 Sven Beiersdorf (63)
Krete 12
242o Eutin-Fissau

Schüler; Mathematik und Englisch
Kartenleser

93 Karsten Beierlein (62)
Breierspfad 83
46oo Dortmund-Wambel
Tel. o231/513o27

Schüler; Elektrotechnik
Kartenleser, Drucker, opt. Lesestift, Memorys, ROMs
-
H: Modellbau (Schiffe)

94 Martin THIELEN (48)
Friedr.-Ebert-Str. 351
415o Krefeld
Tel.: o2151/599512

Studienrat; Mathematik & Physik
Kartenleser, Drucker
Pascal, Basic
H: Foto(color),Musik(Piano),Jazz,Schreinern(Möbel)

95 Elmar Anthon
Dickhardstr. 36
1ooo Berlin 41

96 Guy Poupart
Neugartenstr. 1a
Vianden/Luxembourg

97 Thomas Schreiber (55)
Sonnenstieg 13
34oo Göttingen
Tel. o551/796782

Elektroinstallateur; Schüler, Elektrotechnik, Mathe
Kartenleser, Drucker, Memorys
-
H: HiFi,Rennradfahren,Briefmarken(DR),Mathe,Freundin

---

99 Gustav Wiesel
Goldbachstr. 13
555o Bernkastel-Anden

1oo Roland Roth (53)
Berlinerstr. 123
1ooo Berlin 31
Tel. o3o/8226878

Student; Elektrotechnik
H: Windsurfing, Filmen, Skilanglauf

1o1 Samuel Martin (59)
Homburger Landstr. 767
6ooo Frankfurt 56
Tel. o611/5o72322

Student; Physik
Kartenleser, Memory
Fortran IV
H: Fotografieren, Briefmarken, Astronomie

1o2 Reiner Rosin
Peter-Spahnstr.4
6227 Winkel

Schüler

1o3 Karl Frajo-Apor (44)
Innstr. 44 / Neurum
A-6o2o Innsbruck
Tel. oo43/5222/619372

Ingenieur; Bauwesen (grad.)
Kartenleser,Drucker,Module,Akku,Math-ROM
Basic
H: Reisen, klass. Musik, Lesen

1o4 Lutz Eckert
Klingerstr. 36
8ooo München 7o
Tel. o89-7691993

Dipl.-Ingenieur

1o5 **Walter Bierwirth** **Heidjerhof 1** **3ooo Hannover 61** **Tel. o511/5356-6o29**	**Student; Elektrotechnik** **Kartenleser, Drucker, Mathe ROM, Memorys** **Fortran** **H: Fotografie, SW-Entwicklung, Spiele**
1o6 Frank Schwellinger Neuwiesenstr. 7 7535 Stein Tel. -	Schüler; Mathematik, Physik Kartenleser, Memory Pascal, Basic H:Kino(SF-Filme),Briefmarken,Bowling,Mathe(Knobel.)
1o7 Hans Meiler (41) c/o Glas-Schöninger KG Am Forst 7 / 848o Weiden Tel. o961/21276 bü: 31o55	Prokurist Kartenleser,Drucker,Memorys Basic Modellbahn
1o8 Manfred Engelskirchen (51) Manggasse 2 8o7o Ingolstadt Tel. o841/2581	Filmkaufmann Drucker, Memory - H: Schach
1o9 Jürgen Spiegel Im Sichler 21 635o Bad Nauheim	
11o Michael Semjan Margarethenstr. 7 6ooo Frankfurt 9o	
111 Dr. Heinrich M. Dreyer Postfach 25o148 85oo Nürnberg 25 Tel. o911/593962	Andragoge; Arbeitswissenschaften - Pascal Heimschach
112 Jürgen Holm / bei Hißner (59) Georg-Herwegh-Str. 4 285o Bremerhaven Tel. o471/24692	Student; Physik Kartenleser,Drucker,Memorys H: Musik, Elektronik
113 Heinrich Dörnemann (49) Braunschweigstr. 61 42oo Oberhausen 11 Tel. o2o8/62294	E.Ing.; Medizin-Technik Drucker Basic H: Med-Technik, Tennis
114 Ulrich Iber (5o) Gerresheimer Landstr. 45 4oo6 Erkrath Tel. o211/2o2794	Fluglotse Kartenleser, Drucker, Strichcodeleser, Memorys H: Musik, Reisen
115 Günther Götter (55) Küstrinerstr. 2 753o Pforzheim Tel. o7231/6157o	Student; Feinwerktechnik Kartenleser, Memory Algol 6o, Basic H: Fotografieren, Sport, Musik
116 Ralf Pfeifer Rubensstr. 5 5ooo Köln 5o Tel. o221/352o34	Schüler Kartenleser,Drucker,Memorys,Math-ROM Basic,Pascal H: Wein,Weib und Gitarre
117 Frank Altensen (61) Uhlandstr. 9 6365 Rosbach 3 [illegible]	
118 Heribert Will Grüner Platz 3e 334o Wolfenbüttel	

119 Heinz Rottländer
Königsberger Str. 6
5270 Gummersbach 1
Tel. o2261/66o27

12o Olaf Gursch (49)
Grüntenstr. 29b
1ooo Berlin 42
Tel. o3o/7o32244

Studienrat; Mathe, Physik
Kartenleser
Fortran, Basic, (Pascal)
H: Fotografie, Briefmarken

121 Rudolf Gaier
Erlenweg 7
753o Pforzheim
Tel. o7231/61816

122 Eckhard Müller
Michael-Schnabrich-Str. 9
35oo Kassel

123 Hans-Armin Berk
Lindenstr. 58
638o Bad Homburg
Tel. -

Kaufmann
Kartenleser,Drucker,Finanzmodul
Basic,Pascal

124 Thomas Kellner
Oppenhoffallee 34
51oo Aachen

125 Dr. Wolfgang Fritz (41)
Kronenstr. 34
75oo Karlsruhe 1
Tel. o721/694682

Dipl. Physiker; theor. Physik
Mathe & Standard ROM, Memorys
PL/1, Fortran, Algol, Pascal, APL

126 Heinz-Jürgen Voss
Neustr. 2
4352 Herten

127 Rudolf Kurbasik
Hauptwachstr. 9
86oo Bamberg
Tel. o951/271o1 —büro: 23853

Steuerberater; Dipl.-Kaufmann
Kartenleser,Drucker,opt.Lesestift,Memorys,ROMs
Basic

128 Klaus G. Steger
Gertrud-Herz-Str. 1
872o Schweinfurt
Tel. o9721/45627

Student; Informatik (Technik)
Kartenleser, Memorys
Basic, Pascal
H: Radrennen, HiFi, Alternativtechnik

129 Johannes Schu (55)
Mauthäuslstr. 9
8ooo München 7o
Tel. -

Dipl. Mathematiker
Kartenleser, DDMs
Cobol,Basic,Algol,Fortran

13o Volker Lang (54)
Mühlfeldstr. 3
6222 Geisenheim-Marienthal
(Dürerstr.45/71o3 Schwaigem)

Weinküfer
Kartenleser,Drucker, Memorys
Basic,Fortran IV
H:Bergsteigen,Fotografieren,Luftrettung,Modellfl.

131 Peter Hübsch (43)
Bergstr. 1o
2447 Heiligenhafen
Tel. o4362/1425

Apotheker
Kartenleser
- (Computerschach
H: Fotografie, HiFi, Filmen+Vertonen, Aquaristik

132 Hans-Jürgen Götz (57)
Rietheimerstr. 6
773o VS-Villingen
Tel. o7721/4197

Student; Feinwerktechnik
Kartenleser, Drucker, Mathe-ROM, Memorys
Fortran, Basic, Assembler, Maschinenspr. TMS99oo
H: Fotografieren, Filmen, Elektronik, Skifahren

133 Peter Schwall (47) Tiefenbacher Str. 4 6953 Gundelsheim Tel. o6269/1941	**Bauing. (grad.); selbst.** **Kartenleser, Drucker, Memorys -** **- (Besitze HP-97 mit Statikprogrammen)** **H: Hauskatzen, Gitarrespielen, Judo**
134 Peer Raddatz Tegelsbarg 65 2ooo Hamburg 65	
135 Günter Mohr (36) Henri-Spaak-Str. 24 53o5 Alfter Tel. o228/ 646261	Beamter Kartenleser, Memorys Basic H: Statistik, Tischtennis (HP-67 & TI 59)
136 Reinhard Nickisch Seester 3o b 22o1 Kurzenmoor	
137 Jörg Bendixen (61) Fritz-Flinte-Ring 95 2ooo Hamburg 6o Tel. o4o/63o661o	Schüler - Pascal, APL, Basic H: Tischtennis, Schach
138 Klaus Muschlin Hebelstr. 6 7743 Furtwangen	
139 Thomas Voit (58) Gistlstr. 36 8o23 Pullach	Student; Jura Kartenleser, Drucker Basic
14o Jörg Honsbrok Bergerstr. 118 4o5o Mönchengladbach 1 -	Schüler; Mathematik, Informatik, Physik - Basic,Pascal,Z-8o Assembler H: Mathematik,Heimcomputer,Science Fiction
141 Harald Wienheck Gr. Rainstr. 96 2ooo Hamburg 5o Tel. o4o/59o95o5	Dipl. Biologe Kartenleser,Drucker,Strichcodeleser, ROMs,Memorys H: Fischzucht,Motorflug,Fotografie
142 Hans-O. Carstensen (38) Dorfstr. 42 2331 Barkelsby Tel. o4351/81621	Bau-Ing. (grad.) Kartenleser,Drucker, Module
143 Hartmut Hansen Haarmoor 17 2332 Kosel Tel. o4354/4oo	Student; Bauingenieurwesen Kartenleser, Drucker, Memorys Basic, Fortran
144 Rudolf Siegenthaler Trieschweg 29 CH5o33 Buchs b. Aarau Tel. o64/242425	Prokurist/Kaufmann Kartenleser,Memorys H: SChach
145 Bernd Trilling Brechtingstr. 11 5768 Sundern 2	
146 Gerald Krampe (58) Bern 1 5653 Leichlingen 2 Tel. o2174/3473	Student, E-Technik Kartenleser,Drucker,Memorys,Math-ROM Fortran IV H:Musik(Gitarre),Lesen(Philosophie),Wandern

147 Bernhard Sickmann (4o)
Königskämpe 2
443o Steinfurt 2
Tel. o2552/6o344

Buchhalter
Kartenleser,Drucker,Strichcodeleser,ROMs,Memorys

H: Segel- und Motorflug

148 Heiko Vogeler (62)
Altenrather Str. 14o
52o4 Lohmar 1
Tel. o2246/4882

Schüler, Mathematik, Englisch
Memorys
Basic,Pascal
H:Klaviersp.,klass.Musik,Volleyball,Motorräder

149 Stefan Dräger (63)
Roeckstr. 52
24oo Lübeck
Tel. o451/34311

Schüler
Kartenleser, Drucker, Mathe-ROM, Memorys
Basic (Rechner-Serien-Nr. 1949)
H: Heizungstechnik, Auto, Politik

15o Herbert Knoll
Postfach 147 / Weißenbg. Str. 22
883o Treuchtlingen

151 Christian Lotter
Alexanderstr. 37 / App.4o9
61oo Darmstadt

152 Herbert Platzek
Lisztstr. 5
4o47 Stürzelberg

153 Otmar Harsch
Hauptstr. 165
6236 Eschborn

154 Rainer Tombergs
Noldenkothen 1a
4o3o Ratingen 1

155 Wolfgang Baltes
Guardinistr. 143
8ooo München 7o

156 Lutz Oehler
Cheruskerstr. 19 / Hof links
1ooo Berlin 62

157 Holger Pochert
Goethestr. 57
1ooo Berlin 49

158 Hartmut Wüst
Schwarzenholzweg 1o
663o Saadonis 3

159 Thomas Lindner
Aachener Str. 9 / 3
8ooo München 4o

16o Andreas Kaatz
Emser Str. 113
1ooo Berlin 44

161 Klaus Bartheld
Noeltestr.2o
3ooo Hannover 91

162 Harry Schmidt
Zweibrückenstr. 25
4ooo Düsseldorf 12

163 Hauke Tetens
Friedrich-Ebert-Str. 16
7218 Trossingen
Tel. o7425/6561

164 Marius Heyn
Postfach 22o
7753 Allensbach 1

165 Stefan Trachsler
Haldenstr. 1
CH-828o Kreuzlingen

166 Martin Bueeler
Waffenplatzstr. 8o
CH-8oo2 Zürich

167 Hans-Jürgen Zebisch
Ludwigstr. 2
6146 Alsbach

168 Christa Kurzawski
Petunienweg 1
6382 Friedrichsdorf

169 Hermann-Josef Thomas
Ärmelgasse 3
6531 Waldlaubersheim

170 Günther Buhle
Borsdorfer Weg 18
6478 Nidda 1

171 Christian Krause
Thorwaldsenstr. 19
1ooo Berlin 41

172 Jürgen Menke
Köhlerweg 9
6233 Kelkheim-Rupph.
Tel. o6174/61142

173 Thomas Knauf
Rebenring 63 W31232
33oo Braunschweig

174 Rainer Grzeschkowitz
Inselwall 13
33oo Braunschweig

175 Heinz-Jörg Plegge
FH Dortmund - FB Nachrichtent
Sonnenstr. 96
46oo Dortmund 1

176 Anton Kieninger
Sebastianstr. 8
7o8o Aalen-Waldhausen

177 Andreas Reinhoff
Turmstr. 8
466o Gelsenkirchen-Buer

178 Hans Zupp
Subbelrather Str. 3o
5ooo Köln 3o
Tel. o221/517372

179 Thomas Maldinger
Mülheimerstr. 122
42oo Oberhausen 1

18o Günter Rössl
Pelzmannstr. 1 / c/o Walzl
A-8435 Wagna

181 Andreas Hartwig
Alexanderstr. 265
29oo Oldenburg

182 Bernd Neumann
Alexanderstr. 265
29oo Oldenburg

183 Joachim Spatz
Sachranger Str. 26
8ooo München 9o

184 Peter Penner
Goethestr. 27
4817 Leopoldshöhe

185 Andreas Bußmann
Am Wehr 65
4358 Haltern 5

186 Fred Huber
Badenerstr. 76o
CH-8o48 Zürich

187 Klaus-Dieter Mirus
Spessartstr. 11
1ooo Berlin 33

188 Bernd Heine
Riesengasse 25
684o Lampertheim

189 Alexander Ott
Guldenbergstr. 29
7775 Bermatingen

19o Kemal Essen
Hansaring 125
5ooo Köln 1

191 Gunther Hartmann
Postfach 3o6
632o Alsfeld

192 Gert Liebenberg
Dernburgstr. 12
1ooo Berlin 19

193 René M. Salmen
Prof.-Neu-Allee 8
53oo Bonn 3

194 Klaus Seibert
Eichgärtenallee 4
63oo Giessen

195 Reinhard Klier
Heimstättenweg 66
61oo Darmstadt

196 Max Huber (56)
Talstr. 3o
CH-81o2 Oberengstringen

197 Georg Freund
Webergasse 1b
76oo Offenburg

198 Godehard Willmes
Lerchenweg 14
5253 Lindlar

199 Dr. Horst Wunder
Augustastr. 1o
75o5 Ettlingen

2oo Thomas Fischer
Creniner Str. 16
24oo Lübeck

2o1 Detlef Heitmann
Quickborner Str. 98
2o81 Borstel-Hohenwraden

2o2 Reinhardt Pflanz
Mühlberg 8
6431 Breitenbach / a.H.

2o3 Kurt Pree
Egerländerstr. 8
6352 Ober Mörlen

2o4 Wolfgang König
Weingarten 8
586o Iserlohn

2o5 Walter Pieperhoff
Walb. Osth. Wallstr. 1o
477o Soest

2o6 Jean-Francois Barbey
Tempelhofer Damm 52
1ooo Berlin 42

2o7 Heinz Wehner
Elisabethstr. 4o
23oo Kiel 14

2o8 Gerhard Rüthers
Buchenweg 5
5778 Meschede

2o9 Egon Lindtner
Nienkämperstr. 3
1ooo Berlin 37

21o Dieter Heise
Postfach 1443
462o Castrop-Rauxel

211 Johann Gumze
Brandenburgerstr. 4
6368 Bad Vilbel 2

212 Peter Gölzenleuchter
Auf dem Kalwes 163
463o Bochum 1

213 Stefan Henke
Breitenbachstr. 61
62oo Wiesbaden

214 Jürgen Bunzel
Beethovenstr. 6
68oo Mannheim

215 **Michael Sperl (62)**
Lohstr. 20 a
851o Fürth 17
Tel. o911/761129

216 **Ulrich Mager**
Nelkenstr. 3 W 42
63o5 Buseck 1

217 **Ewald Hagedorn**
Zur Fahlen Schlade 6
5983 Balve 1

218 Udo Wollschläger
Woermannstr. 15
4ooo Düsseldorf 13

219 Alfons Nögel
Sachsenweg 1o
447o Meppen

22o Günter Göck
Beim Schloßgarten 7
68o4 Ilvesheim

221 Dieter W.Klage
Comm. of the Europ. Communi
PO Box 2 / NL-1755 ZG Pette
Tel. o2246/6442

222 Hans Werner Heidbrink
Karpatenweg 15
7o31 Ehningen

223 Dr. Heinz-Jürgen Bernhardt
Grüne Twiete 57
2o83 Halstenbek
Tel. o41o1/44775

224 Werner Loibl (32)
Amalienstr. 47
8858 Neuburg
Tel. o8431/1718 büs 3o55

225 Detlef Gehl
Kirchfeldstr. 59
4o3o Ratingen 8
Tel. o21o2/5o217

226 Marco Pelozzi
Eggweg 11
CH3o65 Bollingen

227 Günter H. Merten
Peter-Martens-Weg 2o
219o Cuxhaven 1

228 Ekkehard Barchewitz
Wasserwerkstr. 33
68oo Mannheim 31

229 Hartmut Brücker
Hauptstr. 58
694o Weinheim

23o **Helmut Seuffert (55)**
Langweidenstr. 6
6ooo Frankfurt 9o
Tel. o611/ 7[illegible]93444

231 **Jörg Meyer**
Lange Str. 54
2833 Harpstedt

232 Martin Klingenberg
Lohbergstr. 12
3428 Duderstadt 18

233 Ralph Kütemeier
Elsässerstr. 29
29oo Oldenburg

234 Jan Schäfer-Kunz
Tuttlingerstr. 22
7ooo Stuttgart 75

235 Reinhold Elsebach
Adalbert-Stifter-Str. 1o
3558 Frankenberg
Tel. o6451/4727

236 Walter Brandner
Johannagasse 35a / 14
A-1o5o Wien

237 Ulrich Pahmeyer
Jahnweg 15
782o Titisee-Neustadt

238 Michael Wandl
Mentergasse 9/1o
A-1o7o Wien

239 Günter Lehrke
Wiesenstr. 85
7o24 Fildarstadt 1

24o Rüdiger Schultz
Villa Viktoria
CH-94o2 Mörschwil

241 Manfred Mickoteit
Am alten Hospital 9
54oo Koblenz
Tel. o261/38469

242 Walter M. Schütz
Bahnhofstr. 43
CH-87o2 Zollikon

243 Peter Stock
Rienzistr. 1o
7ooo Stuttgart 7o

244 Alois Widmann
Veitsmüllerweg 13
8o5o Freising

Böhmerwaldweg 5
6368 Bad Vilbel

246 Ulrich Graser
Panoramastr. 22
7o16 Gerlingen

247 Friedemann Fehler/ c/o MBB
MBB Abt. RTZ, Postfach 8o116
8ooo München 8o

248 Reiner Mang
Industriestr. 8
6325 Grebenau

249 Christoph Hort
Am Mühlenwäldchen 11
667o St. Ingbert

25o Peter Grützmacher
Im alten Garten 24
55oo Trier

251 Uwe Hommelsheim
Unnauer Weg 5o
5ooo Köln 71

252 Gerd Kretschmann
Christernstr. 39 / PF 448436
28oo Bremen 44
Tel. /452211

253 Klaus Bendfeldt
Bernauer Str. 56
592o Bad Berleburg
Tel. o2751/3867

254 Claus Martin Driever
Claßenstr. 15
51oo Aachen

255 Gerhard Schott
Am Bockzahl 23
6479 Schotten 1

256 Michael Schaale
Florastr. 6
1ooo Berlin 41
Tel. o3o/7914o94

257 Klaus Evers
Post Nusse
2412 Mannhagen

258 Andreas Meyer-Lindenberg (65)
Elliger Höhe 31
53oo Bonn 2
Tel. o228/323255

259 Jean-Paul Fay
Rue J.Peschong 11
L-4938 Baschavage (Luxemb.)

260 Dr.Dr. Horst Marggraf
Berliner Str. 3
1ooo Berlin 27

261 Marcus Möhrmann
Lauterberger Weg 12
2ooo Hamburg 61

262 (Herr) Anne v. Houwelingen
Gen. Spoorstr. 4o5
NL- 3313 AE Dorohecke

263 Thomas Rölz
Windsheimer Str. 431 a
85oo Nürnberg

264 Frank Guschmann
Lützowstr. 2o
1ooo Berlin 3o

265 Lukas Rohr
Dreilindenstr. 39
CH-6oo6 Luzern

266 Michael Gehret
Silcherstr. 15
8944 Grönenbach
Tel. o8334/1oo1

267 Michael Gerhardt
Bünkamp 7o
44oo Münster

268 Olaf Pilling
Hattenheimer Str. 9 b
1ooo Berlin 28

269 Uwe Kremmin
Auf der Gasse 6
59oo Siegen 31

27o Herbert Brähler
Schulstr. 2
6368 Bad Vilbel 1
Tel. o6193-64775

271 Bernd Haug
Am Kirchberg 8
72o1 Seitingen-Oberflacht

272 Werner Grunske
Goldmühle Str. 11
7o32 Sindelfingen

273 H.L. Koot
Louise de Colignylaan 12o
NL- 3136 CS Vlaardingen
Tel. o1o/744915

274 Christian Thiele
Hannoversche Str. 11
3o5o Wunstorf 1

275 [illegible]er Riethenauer
Am Pfädle 3
7519 Gemmingen

276 Joachim Welters
Saarlandstr. 49
63oo Gießen/ Kl. Linden

277 Christian Herr
Rosstr. 63-65
51oo Aachen

278 Albert Damm
Lessingstr. 12
667o St. Ingbert
Tel. o6894/6329

279 Helmut Loepp
Kleiberweg 17
46oo Dortmund 3o
Tel. o231/413655

28o Jürgen Moeck
Dellplatz 7 / 12o
41oo Duisburg 1

281 Jürgen Steinbrecht
Am Trieb 15 W 7.3
6o78 Neu-Isenburg

282 Uwe Scharfenberg
Lensahnerstr. 4
2433 Cismar

283 Ulrich Jansen
Mühlenstr. 16
514o Erkelenz

284 J. Walenta
Gratstr. 12
CH-8472 Seuzach

285 Michael Haase
Ringstr. 37
79o4 Erbach 1

286 Peter Astheimer
Wilhelmstr. 13
6o9o Rüsselsheim

287 Lutz Löhnert
Bocholtzer Str. 28
51oo Aachen-Vetschau

288 Johannes Palic
Gemseneck 2o
2ooo Hamburg 54
Tel. o4o/546311

289 Hans-Joachim Tepper
Bueltenweg 73
3300 Braunschweig

290 Klaus Weber
Leher Heerstr. 102 / 11a
2800 Bremen 33
Tel.0421/231942

291 Werner Fritz
Pachernhauptstr. 66
A-8042 Graz

292 Franz Brunschwiler
Adlerstr. 3
CH-4133 Pratteln

293 Klaus D. Veil
Hansengelstr. 26
7891 Hohentengen

294 Peter Hess
Durlacherstr. 41
6800 Mannheim 81
Tel. 0621/896248

295 Holger Suhr
Kaiserstr. 82
2300 Kiel 14

296 Fritz Nickolai
Salzlände 16
A-8706 Leoben

297 Michael Grandau
Seilandstr. 7
4714 Selm
Tel. 02592/4333

298 Edgar Sieberg
Bummelberg 35
4600 Dortmund 1

299 Jens Schleusener
Burgstr. 9
3400 Göttingen

300 Heinrich Beyer
Essener Str. 26
4800 Bielefeld 14

301 Winfried Kadach
Thorner Str. 14
3406 Bovenden 1
Tel. 0551/81267

302 Wilhelm Handler
Semmelweisgasse 29
A-1210 Wien

303 Uwe Letas
Zedeliusstr. 33
2940 Wilhelmshaven

304 M. Steffens
Odenwaldstr. 6
5047 Wesseling

305 Burghard Schatz
Fischenicherstr. 37
5000 Köln 50

306 Siegfried Geipel
Roggensteiner Allee 240
8031 Eichenau

307 Hans-Egon Ramundt
Truderingerstr. 154
8000 München 82
Tel. 089/436832

308 Edzard Lueg
Lettow-Vorbeck-Str. 2
2870 Delmenhorst

309 Richard Gaschler
Ludwigsburger Steige 52
7141 Aldingen/Remseck

310 Knut Menzel
Schubertplatz 12
4930 Detmold

311 Bernhard Leiner
Blumenstr. 27
6781 Vinningen

312 Annegret Jäger
Zum Grossen Busch 49
5600 Wuppertal 17
Tel. 02128/87264

313 Rolf Leuenberger
Niesenweg 8
CH-3012 Bern

314 Otfried Werner
Myliusstr. 59
6000 Frankfurt 1

315 Michael Willer
Wallstr. 28a
2944 Wittmund 1

316 G.M. Günther
Hermanstr. 33 a
8900 Augsburg 1

317 P.R.J.J. Crombeen
Campuslaan 27-206
7522 NC Enschede / Holland

318 Wolfgang Lesemann
Grundstr. 28 a
4930 Detmold 17

319 Ulrich Dahnz
Karlstr. 7
6352 Ober-Mörlen

320 Hans Peter Duwe
Christopher Probst Str. 16-705
8000 München 40

321 Hans Michel
Bündnerstr. 46
CH-4055 Basel

322 Wolfgang Zingsem
Winkelnerstr. 55
4050 Mönchengladbach 6

323 Harald Kutz
Steinstr. 16
4100 Duisburg 12
Tel. 0203/444136

324 Thomas Kläger
Gattikonerstr. 98
CH-8136 Gattikon

325 Michael Kannenberg
Madenstr. 18
7000 Stuttgart 75
Tel. 0711/473592

326 Ronald Jore
Bornimer Str. 18
1000 Berlin 31
Tel. 030/8914656

327 Frank Lange
Althoffstr. 20
4600 Dortmund 1

328 Joachim M. Kiefer
Großgerstenfeld 1 a
6635 Schwalbach - Ensdorf
Tel. 06831/52569

329 Gudrun Rahn
Breite Str. 4
3400 Göttingen

330 Gerhard Bültemann
Fährstr. 60
2800 Bremen 1

Niels Nöhren
Kielort 16
2000 Norderstedt
F: (040) 524 28 88
Pschk Hmb 299375-201

Norderstedt, 28.[illegible]1980

Herrn
Oliver Rietschel
Hewlett-Packard
Anwender-Club
Postfach 373

2420 Eutin

Lieber Oliver, liebe Clubfreunde!

In einem der Weihnachts-Info-Hefte erschien das Programm PFZ von Christof Born. Ein Programm, das jede ganze positive Zahl in ihre Primzahlfaktoren zerlegt. Ich möchte zu diesem Programm einen Verbesserungsvorschlag machen.

Durch das beigelegte Listing möchte ich zeigen, wie man das Programm verbessern und um rund 35% nämlich von 113 auf 73 Programmschritte verkürzen kann.

Im Programm ist ein wirklich echter Fehler enthalten. Wer es in der ursprünglichen Fassung übernehmen und verwenden möchte, wird zu seienm Erstaunen feststellen müssen, daß das Programm die Zahlen 25 und 49 als Primzahlen ausweist. Der Fehler liegt beim Programmschritt 52. Der Vergleichsbefehl "X)Y?" allein reicht hier nicht aus. Vielleicht ist Christof ein Gedankenfehler unterlaufen. Verwendet man jedoch zusätzlich einen 2. Vergleichsbefehl, nämlich "X≠Y?" (wie auch schon Matthias Grabiak in seinem Beitrag vom 14.09.1980 angeregt hat), werden auch die Zahlen 25 und 49 richtig in ihre Primzahlfaktoren zerlegt.

Übrigens, ein Hinweis für alle, die es vielleicht noch nicht wissen. Ein Enter-Befehl vor einem RCL-Befehl ist überflüssig, weil bei "RCL" der automatische Stack-Lift sowieso in Gang gesetzt wird. Noch etwas fällt mir ein. Die Befehlsfolge:
Rcl Ø1, RCL Ø2, /, RCL Ø1, RCL Ø2 /, INT, läßt sich verkürzen auf: RCL Ø1, RCL Ø1, RCL Ø2, /, INT.

Ich hoffe, daß ich mit meinem Beitrag nützlich sein konnte.
Ich wünsche weiterhin

Happy Programming.

Programm PFZ von Christof Dorn. Optimiert von Niels Nöhren.

```
01 LBL PFZ
02 FIX Ø
03 CF ØØ
04 "Z=?"
05 PROMPT
06 STO ØØ
07 STO Ø1
08 INT
09 LASTX
10 X≠Y?
11 GTO Ø3
12 2
13 STO Ø2
14 X)Y?
15 GTO Ø3
16 X=Y?
17 GTO Ø2
18 X()Y
19 8
20 SF Ø2
21 X)Y?
22 SF Ø1
23 CLA

24 LBL Ø1
25 RCL Ø1
26 RCL Ø2
27 MOD
28 X=Ø?
29 GTO Ø4
30 FS? Ø1
31 GTO Ø2
32 FS? Ø2
33 3
34 FS? Ø2
35 STO Ø2
36 FS?C Ø2
37 GTO Ø1
38 RCL ØØ
39 2
40 ST+ Ø2
41 /
42 RCL Ø2
43 X)Y?
44 GTO ØØ
45 FS? ØØ
46 GTO Ø1
47 RCL ØØ
48 SQRT
49 X≠Y?
50 X)Y?
51 GTO Ø1

52 LBL Ø2
53 "PRIMZAHL"
54 GTO ØØ

55 LBL Ø3
56 "UNZERLEGBAR"
57 GTO ØØ

58 LBL Ø4
59 SF ØØ
60 CF Ø1
61 RCL Ø2
62 ARCL X
63 ST/ Ø1
64 RCL Ø1
65 1
66 X≠Y?
67 GTO Ø1

68 LBL ØØ
69 CF ØØ
70 CF Ø1
71 CF Ø2
72 AVIEW
73 END
```

```
01♦LBL "KENNER"
02 SF 01
03♦LBL "QTH"
04 AON
05 "1. BUCHST.?"
06 PROMPT
07 XEQ 12
08 INT
09 STO 00
10 "2. BUCHST.?"
11 PROMPT
12 AOFF
13 XEQ 12
14 FRC
15 100
16 *
17 INT
18 STO 01
19 CLX
20 80
21 "ZAHL ?"
22 PROMPT
23 STO 02
24 X<=0?
25 GTO 14
26 X>Y?
27 GTO 14
28 1
29 ST- 02
30 RCL 02
31 10
32 /
33 STO 02
34 INT
35 8
36 /
37 ST- 01
38 RCL 02
39 FRC
40 10
41 *
42 INT
43 5
44 /
45 ST+ 00
46 AON
47 "LETZT.BUCHST?"
48 PROMPT
49 AOFF
50 XEQ 12
51 100
52 *
53 FRC
54 X=0?
55 GTO 14
56 10
57 *
58 STO 02
59 INT
60 30
61 /
62 ST+ 00
63 RCL 02
64 FRC
65 10
66 *
67 48
68 /
69 ST- 01
70 FC?C 01
71 GTO A
72 RCL 00
73 STO 03
74 RCL 01
75 STO 04
76 GTO "QTH"
77♦LBL A
78 RCL 00
79 RCL 03
80 -
81 COS
82 RCL 01
83 COS
84 *
85 RCL 04
86 COS
87 *
88 RCL 04
89 SIN
90 RCL 01
91 SIN
92 *
93 +
94 ACOS
95 111.12
96 *
97 FIX 2
98 CF 25
99 RND
100 CLA
101 ARCL X
102 "⊢ KM"
103 AVIEW
104 STOP
105 GTO "QTH"
106♦LBL 12
107 ASTO X
108 SF 25
109 XEQ IND X
110 FS? 25
111 RTN
112 "*"
113 ARCL X
114 ASTO X
115 SF 25
116 XEQ IND X
117 FC? 25
118 GTO 14
119 RTN
120♦LBL 14
121 TONE 9
122 "NEUE EINGABE"
123 AVIEW
124 PSE
125 GTO "QTH"
126♦LBL "Z"
127 50.66
128 RTN
129♦LBL "Y"
130 48.65
131 RTN
132♦LBL "X"
133 46.64
134 RTN
135♦LBL "W"
136 44.63
137 RTN
138♦LBL "V"
139 42.62
140 RTN
141♦LBL "U"
142 40.61
143 RTN
144♦LBL "T"
145 38.6
146 RTN
147♦LBL "S"
148 36.59
149 RTN
150♦LBL "R"
151 34.58
152 RTN
153♦LBL "Q"
154 32.57
155 RTN
156♦LBL "P"
157 30.56
158 RTN
159♦LBL "O"
160 28.55
161 RTN
162♦LBL "N"
163 26.54
164 RTN
165♦LBL "M"
166 24.53
167 RTN
168♦LBL "L"
169 22.52
170 RTN
171♦LBL "K"
172 20.51
173 RTN
174♦LBL "*J"
175 18.5033
176 RTN
177♦LBL "*I"
178 16.49
179 RTN
180♦LBL "*H"
```

Jürgen Spiegel, [illegible], Dez. 1980

```
181 14.4811
182 RTN
183♦LBL "*G"
184 12.4713
185 RTN
186♦LBL "*F"
187 10.4615
188 RTN
189♦LBL "*E"
190 8.4535
191 RTN
192♦LBL "*D"
193 6.4455
194 RTN
195♦LBL "*C"
196 4.4353
197 RTN
198♦LBL "*B"
199 2.4251
200 RTN
201♦LBL "*A"
202 .4131
203 RTN
204 END
```

HP 41c –
Programm für Entfernungsbestimmung nach QTH-Kennern.
(Amateurfunk)

Size 005 ; 568 Bytes f. Programm

Tastenbelegung USER : KENNER ;
QTH

Mit "KENNER" wird das Programm gestartet. Der Rechner fragt nach den eigenen Daten:

QTH-kenner 1. Buchstabe
2. Buchstabe
Zahl
Letzt. Buchstabe

Danach wird der eigene Standort in geogr. Länge u. Breite ermittelt und in Register 03 u. 04 abgespeichert.

Dann fragt der Rechner nach dem QTH-kenner des Zieles und ermittelt die Entfernung in km.
(Die Daten des Zielortes werden in R 00 und 01 gespeichert)

Für weitere Entfernungsermittlungen wird das Programm mit Taste "QTH" gestartet, da die eigenen Daten in R03/04 bereits gespeichert sind.

Batteriesatz des Thermodruckers

Früher oder später wird der aufladbare Batteriesatz des Thermodruckers seinen Geist aufgeben; bei mir war eine der vier Batterien schon nach kurzer Zeit hinüber.

Bei den erfahrungsgemäß nicht gerade bescheidenen Preisen von HP hier nun ein Tip zum Sparen: Vier Ni-Cd-Accumulatoren des Typs RSH 1,2 der Firma VARTA passen in das Batteriefach des Druckers. Sie sind sogar mit Lötösen versehen, was die Montage vereinfacht. Es empfiehlt sich allerdings, die Zuleitungsdrähte zu den Kontaktfedern des Druckers dort nicht gerade da anzulöten, wo sie beim serienmäßigen Batteriesatz der Kontaktgabe dienen, falls man später doch mal wieder einen solchen verwenden will oder muß.

Der Preis der VARTA-Batterien liegt bei ca. DM 10. Sie sind beispielsweise erhältlich bei der Fa. Walter Kluxen, Nordkanalstr. 52, 2000 Hamburg 1 unter der Bestellnummer 90 1660

Klaus Werner Hoenow

Ergänzungsvorschlag zum Programm MULTI

Das ausgezeichnete Plotprogramm MULTI aus "prisma" vom November 1980 könnte m. E. noch ganz gut eine Abszissenbeschriftung gebrauchen:

Man fügt zwischen Zeile 319 und END die untenstehenden Schritte ein.

Nach Ausdruck des vollständigen Funktionenverlaufs wird durch XEQ ABSZ eine Abszissenskala erzeugt.

Die Zeilen 327 (Format der Abszissenbeschriftung) und 334 (Abstand der beschrifteten Abszissenpunkte) können vom Benutzer nach eigenem Geschmack verändert werden.

Hamburg, den 24.11.1980 Klaus Werner Hoenow

```
320 RTN              339 ARCL 07
321◆LBL "ABS         340 ACA
Z"                   341 1
322 ADV              342 SKPCHR
323 XEQ 13           343◆LBL 17
324 ADV              344 SF 12
325 0                345 45
326 STO 00           346 ACCHR
327 SCI 2            347 ADV
328◆LBL 16           348 CF 12
329 RCL 08           349 RCL 09
330 RCL 07           350 ST+ 07
331 X>Y?             351 1
332 GTO 18           352 ST+ 00
333 RCL 00           353 GTO 16
334 5                354◆LBL 18
335 MOD              355 XEQ 13
336 Y≠0?             356 ADV
337 GTO 17           357 BEEP
338 CLA
```

Unterprogramm FORMAT

Das Unterprogramm FORMAT "rettet" ein einmal manuell oder per Programm eingestelltes Anzeigeformat über Formatveränderungen (beispielsweise für Index-Ausdruck o.ä.) hinweg.

Nach Ausführung von FORMAT, das nur den Stack benutzt, steht im Y-Register die Anzahl der Nachkommastellen des zu rettenden Formats; im X-Register steht eine Kennzahl, nämlich 0 für FIX, 1 für SCI und 2 für ENG. Beide Ziffern können in beliebige Register zur späteren Verwendung abgespeichert werden, um dann das ursprüngliche Format wieder zu rekonstruieren. Letzteres muß durch bis zu 3 vom Benutzer zu erstellende kurze Unterprogramme geschehen, wie das folgende Beispiel zeigt:

In diesem, allerdings sehr trivialen Beispiel sollen die Vielfachen N * PI für N = 1...5 gebildet werden, wobei N ohne Nachkommastellen, das jeweilige Vielfache in einem vorher manuell eingestellten Format angezeigt werden sollen. Im Beispielprogramm N*PI wird dieses Format in jedem Schleifendurchlauf durch Zeile 12 zerstört; es muß also hinübergerettet werden. Das geschieht durch die Zeilen 02...05, wobei die Kennzahl des Anzeigemodus nach R00, die Anzahl der Nachkommastellen nach R01 gespeichert wird. Die Unterprogramme 00 (Zeile 25), 01 (Zeile 28) und 02 (Zeile 31) stellen dann das ursprünglich gewünschte Anzeigeformat wieder her *(aufgerufen durch Zeile 18)*.

Hamburg 73, den 01.12.1980 Klaus Werner Roenow

```
 01*LBL "FOR
MAT"
 02 39.035
 03 ENTER↑
 04 0
 05 ENTER↑
 06 1
 07*LBL 01
 08 FS? IND
Z
 09 +
 10 FS? IND
Y
 11 LASTX
 12 2
 13 *
 14 DSE Z
 15 GTO 01
 16 CLX
 17 FS? 40
 18 RTN
 19 1
 20 +
 21 FC? 41
 22 RTN
 23 1
 24 +
 25 RTN
 26 END
```

```
 01*LBL "N*P
I"
 02 XEQ "FOR
MAT"
 03 STO 00
 04 RDN
 05 STO 01
 06 1.005
 07 STO 02
 08 CF 29
 09*LBL 03
 10 RCL 02
 11 INT
 12 FIX 0
 13 CLA
 14 ARCL X
 15 "⊢: "
 16 PI
 17 *
 18 XEQ IND
00
 19 ARCL X
 20 AVIEW
 21 ISG 02
 22 GTO 03
 23 ADV
 24 RTN
 25*LBL 00
 26 FIX IND
01
 27 RTN
 28*LBL 01
 29 SCI IND
01
 30 RTN
 31*LBL 02
 32 ENG IND
01
 33 END
```

```
        FIX 3
  XEQ "N*PI"
1: 3.142
2: 6.283
3: 9.425
4: 12.566
5: 15.708

        SCI 2
  XEQ "N*PI"
1: 3.14E0
2: 6.28E0
3: 9.42E0
4: 1.26E1
5: 1.57E1
```

Programm MSMIMA SIZE 010

MSMIMA ermittelt außer Mittelwert und Standardabweichung einer Folge von Werten auch noch deren minimalen und maximalen Wert.

Start des Programmes durch XEQ MSMIMA. Es wird Σ+(Σ-) zur Symbolisierung der Eingabe angezeigt. Diese erfolgt also wie gewohnt durch Σ+, Löschung eines versehentlich falsch eingegebenen Wertes durch Σ-.
Nach Eingabe aller Werte erfolgt durch Drücken von B nacheinander die Anzeige von Mittelwert, Standardabweichung, Minimum und Maximum, und zwar bei eingeschaltetem Drucker automatisch, sonst nach Drücken von R/S.
Minimum und Maximum sind in den Speichern R00 bezw. R01 verfügbar.

Hamburg 73, den 01.12.1980 Klaus Werner Hoenow

```
LBL"MSMIMA
END
105 BYTES

01◆LBL "MSM
IMA"
02 ΣREG 04
03 CLΣ
04 9 E99
05 STO 00
06 CHS
07 STO 01
08 CLST
09 " Σ+ (Σ-
)"
10 SF 27
11 PROMPT
12◆LBL A
13 RCL 00
14 STO 02
15 X<>Y
16 X<Y?
17 STO 00
18 RCL 01
19 STO 03
20 X<>Y
21 X>Y?
22 STO 01
23 0
24 X<>Y
25 Σ+
26 RTN
27◆LBL a
28 RCL 02
29 STO 00
30 RCL 03
31 STO 01
32 0
33 R↑
34 Σ-
35 RTN
36◆LBL B
37 "M="
38 MEAN
39 ARCL X
40 AVIEW
41 FC? 55
42 STOP
43 "S="
44 SDEV
45 ARCL X
46 AVIEW
47 FC? 55
48 STOP
49 "MIN="
50 ARCL 00
51 AVIEW
52 FC? 55
53 STOP
54 "MAX="
55 ARCL 01
56 AVIEW
57 ΣREG 11
58 END
```

HP-41 C

Programmpaket MATRIZENRECHNUNG

Dieses Programmpaket umfaßt mehrere Unterprogramme zur Matrizenrechnung, die so aufgebaut sind, daß eine oder mehrere Matrizen beliebiger Reihen- und Spaltenzahl (soweit es der Speicherplatz zuläßt) den Operationen des Matrizenkalküls unterzogen werden können. Zu diesem Zweck arbeiten die einzelnen Unterprogramme mit einer dynamischen Speicherbereichszuweisung, die für jede Matrix den benötigten Speicherbedarf folgendermaßen festlegt:

1. Speicherung

   Gegeben sei die m*n-Matrix A mit den Elementen a_{ij} $(i=1,...,m;\ j=1,...,n)$. Die Speicherung beginnt mit dem Register R_S, wobei S die Startadresse für die Matrix A ist, mit der sie von allen Unterprogrammen aufgerufen wird. Dabei muß $S \geq 11$ gewählt werden, da die Register R_{00} bis R_{10} für Zwischenspeicherungen innerhalb der einzelnen Unterprogramme benutzt werden.

   Die Matrix A nimmt insgesamt m*n+3 Register in Anspruch, die folgendermaßen belegt sind:

   R_S : "A"
   R_{S+1} : m
   R_{S+2} : n
   $R_{S+3}...R_{S+2+m*n}$: a_{ij} $(i=1,...,m;\ j=1,...,n)$

   R_S enthält also den Namen der Matrix (max. 6 ALPHA-Zeichen), R_{S+1} die Zeilenzahl und R_{S+2} die Spaltenzahl. Die folgenden Register R_{S+3} bis $R_{S+2+m*n}$ enthalten die m*n Matrixelemente a_{ij} zeilenweise fortlaufend.

2. Adressierung

   Die Registeradresse r eines Matrixelementes a_{ij} der m*n-Matrix A, deren Startadresse S ist, errechnet sich nach dem oben Gesagten zu

   $$r = S + 2 + (i - 1) * n + j$$

   Diese Adressenberechnung wird von MTAIJ durchgeführt, das daher von fast allen Unterprogrammen dieses Programmpaketes

benötigt wird. MTAIJ benutzt nur den Stack.

3. Zwischenspeicherung

Die Register R_{00} bis R_{10} bleiben für die verschiedenen Unterprogramme als Zwischenspeicher reserviert, wobei der Speicherplan, außer für MTINV, so aussieht:

R_{ALPHA}	: Name der Ergebnismatrix
R_{00}	: Konstante oder sonstige Hilfsgröße
R_{01}	: Startadresse der 1. Matrix
R_{02}	: " " 2. "
R_{03}	: " " Ergebnismatrix
R_{04}	: Element der 1. Matrix
R_{05}	: " " Ergebnismatrix
R_{06}	: 1. Schleifenindex
R_{07}	: 2. "
R_{08}	: 3. "
R_{09}	: Re-Initialisierung des 2. Schleifenindex
R_{10}	: " " 3. "

Für MTINV werden die Register R_{00} bis R_{08} benutzt.

4. Unterprogrammaufruf

Bei Aufruf der einzelnen Unterprogramme dieses Programmpaketes müssen gewisse Daten im Stack bezw. im ALPHA-Register vom Benutzer bereitgestellt werden, wie es jeweils im Kopf der einzelnen Listings angegeben ist. Das gilt jedoch nicht für MTAIJ, da die aufrufenden Unterprogramme dies bereits selbsttätig tun.

5. Programmbeschreibungen

5.1 MTAIJ

Dieses Unter-Unterprogramm wurde unter Pkt. 2 erklärt.

5.2 MTEIN

Die Elemente a_{ij} der Matrix A werden im Dialogverkehr vom Benutzer reihenweise eingegeben.

5.3 MTAUS

Die Elemente a_{ij} der Matrix A werden mit Benennung reihenweise (CF 01) oder spaltenweise (SF 01) ausgegeben.

Bei eingeschaltetem Drucker erfolgt die Ausgabe kontinuierlich; sonst wird die Ausgabe nach jedem Element gestoppt und muß mit R/S fortgesetzt werden.

5.4 MT111

Es wird eine m*m-Einheitsmatrix E erzeugt, bei der die Elemente der Hauptdiagonalen Eins sind, alle übrigen jedoch Null.

5.5 MTADD

Dieses Unterprogramm vereinigt die Matrizenaddition mit der Multiplikation mit einem konstanten Faktor:

$$C = A + k * B$$

bezw. $c_{ij} = a_{ij} + k * b_{ij} \quad (i=1,\ldots,m;\ j=1,\ldots,n)$

Für k = 1 ergibt sich die normale Matrizenaddition. Für A = $\underline{0}$ (Nullmatrix) erhält man die Multiplikation von B mit dem konstanten Faktor k. In diesem Fall muß bei Aufruf von MTADD im T-Register des Stack eine Null stehen; die Nullmatrix $\underline{0}$ braucht nicht gespeichert zu sein. Die Ergebnismatrix C darf sowohl einen von A und B verschiedenen Platz in den Registern einnehmen als auch eine von beiden Matrizen ersetzen.

5.6 MTSPU

Die Spur einer m*m-Matrix A ist definiert als die Summe ihrer Diagonalelemente:

$$\mathrm{Spur}(A) = \sum_{i=1}^{m} a_{ii}$$

Sie wird in dem Register gespeichert, das bei Aufruf von MTSPU als Zahl im X-Register steht.

5.7 MTSHF

Dieses Unterprogramm dient zum Kopieren oder Verschieben von Matrizen innerhalb der Register. Beim Verschieben werden die Registerinhalte der "alten" Speicherplätze nicht gelöscht. Eine Verschiebung um weniger als m*n+3 Register ist erlaubt, wobei dann natürlich "alte" Elemente durch "neue" überschrieben werden.

5.8 MTTRA

Dieses Unterprogramm erzeugt die Transponierte einer m*n-Matrix A, nämlich die n*m-Matrix B, die aus A durch

Vertauschen von Reihen und Spalten hervorgeht:

$$b_{ji} = a_{ij} \quad (i=1,\ldots,m;\ j=1,\ldots,n)$$

5.9 MTMUL

Gegeben seien die l*m-Matrix A und die m*n-Matrix B (Spaltenzahl von A = Reihenzahl von B). MTMUL erzeugt die l*n-Produktmatrix C = A * B nach

$$c_{ik} = \sum_{j=1}^{m} a_{ij} b_{jk} \quad (i=1,\ldots,l;\ j=1,\ldots,m;\ k=1,\ldots,n)$$

Im allgemeinen ist A * B ≠ B * A !

5.10 MTINV

Dieses Unterprogramm erzeugt die Inverse $B = A^{-1}$ einer m*m-Matrix A. Es gilt

$$B * A = A * B = E$$

mit E als Einheitsmatrix (siehe Pkt. 5.4). B ist dann natürlich auch eine m*m-Matrix.

Es wird m-mal das Stiefel-Verfahren angewandt mit a_{11} als Pivotelement; nach jeder Pivotisierung werden die neuen Matrixelemente in bestimmter Weise zyklisch vertauscht. Näheres zu diesem Verfahren z. B. bei G. Venz: "Lineare Algebra für programmierbare Taschenrechner", Oldenbourg-Verlag 1980, S. 24 ff.

Außer der Bedingung, daß A quadratisch und nicht-singulär sein muß (siehe Lehrbücher über Matrizenrechnung), darf bei diesem Verfahren kein Element der Hauptdiagonalen Null sein. Wenn dieses Unterprogramm zusammen mit MTMUL zur Auflösung linearer Gleichungssysteme verwendet werden soll, läßt sich diese Zusatzbedingung meist durch Umstellung der Gleichungen erreichen.

Nach Abarbeitung von MTINV hat $B = A^{-1}$ die Elemente von A in den Speichern überschrieben!

. Beispiel

Die 3*3-Matrix

$$A = \begin{bmatrix} 4 & 7 & 2 \\ -6 & 1 & 0 \\ 2 & 7 & 3 \end{bmatrix}$$

soll folgenden Operationen unterzogen werden:

6.1 Einlesen von A (MTEIN)

6.2 Reihenweises Ausgeben von A (MTAUS)

6.3 Kopieren von A nach B (MTSHF)

6.4 Bilden von $C = A^{-1}$ (MTINV)

6.5 Spaltenweises Ausgeben von C (MTAUS)

6.6 Bilden von $E = C * B$ (MTMUL)

6.7 Reihenweises Ausgeben von E (MTAUS)

6.8 Bilden und Ausgeben der Spur von E (MTSPU)

Außer den angegebenen Unterprogrammen wird noch MTAIJ benötigt.

Jede Matrix benötigt nach Pkt. 1 3*3+3=12 Register, wobei hier mit R_{11} begonnen werden soll. C nimmt nach Ausführung von MTINV den Platz von A ein, so daß für insgesamt 3 Matrizen, nämlich A, B und E, Speicherplatz reserviert werden muß. Es wird gewählt:

S(A) = S(C) = 11
S(B) = 23
S(E) = 35

Die Spur von E soll auf dem später nicht mehr benötigten Platz R_{11} gespeichert werden.

Um Programmspeicherplatz zu sparen, wird mit Hilfe der RSUB-Funktion des Kartenlesers eine Art "Overlay"-Technik durchgeführt: Es werden zunächst nur die beiden relativ kurzen und häufig gebrauchten Unterprogramme MTAIJ und MTAUS abgespeichert; anschließend folgt das aufrufende Hauptprogramm HAUPT, wie der folgende Katalog und das Listing zeigen:

```
        CAT 1
LBL'MTAIJ
END
25 BYTES
LBL'MTAUS
END
100 BYTES
LBL'HAUPT
END
171 BYTES
.END.
05 BYTES
```

```
 01♦LBL "HAU       20 XEQ "MTS        39 XEQ "MTM
PT"               HF"                UL"
 02 "MTEIN"        21 "MTINV"         40 CF 01
 03 AVIEW          22 AVIEW           41 35
 04 RSUB           23 RSUB            42 XEQ "MTA
 05 "A"            24 "C"            US"
 06 3              25 11              43 "MTSPU"
 07 ENTER↑         26 XEQ "MTI        44 AVIEW
 08 ENTER↑        NV"                 45 RSUB
 09 11             27 SF 01           46 35
 10 XEQ "MTE       28 11              47 ENTER↑
IN"                29 XEQ "MTA        48 11
 11 CF 01         US"                 49 XEQ "MTS
 12 11             30 "MTMUL"        PU"
 13 XEQ "MTA       31 AVIEW           50 FIX 2
US"                32 RSUB            51 ADV
 14 "MTSHF"        33 "E"             52 "SPUR<E>
 15 AVIEW          34 11             ="
 16 RSUB           35 ENTER↑          53 ARCL 11
 17 11             36 23              54 AVIEW
 18 ENTER↑         37 ENTER↑          55 END
 19 23             38 35
```

HAUPT erfüllt folgende Funktionen:

- Steuerung des Aufrufs der Unterprogramme gemäß 6.1 bis 6.6, wobei die für die Unterprogramme benötigten Parameter im Stack bezw. im ALPHA-Register bereitgestellt werden
- Anforderung der benötigten Unterprogramme vom Benutzer mittels RSUB, wobei der Name des Unterprogrammes in der Anzeige erscheint. Der Benutzer hat dann lediglich die zugehörige Unterprogrammkarte in den Kartenleser einzuführen, und das Hauptprogramm setzt die Ausführung automatisch fort.

Auf diese Weise werden zusätzlich zu den für MTAIJ, MTAUS und HAUPT benötigten 43 Registern lediglich noch 27 Speicherregister (Unterprogramm MTINV ist das längste) benötigt, da durch den RSUB-Befehl immer das hinter HAUPT stehende Programm überschrieben wird. Die restlichen Register stehen als Datenspeicher für die Aufnahme der Matrizen (auch größerer!) zur Verfügung. Da für dieses Beispiel 70 Programmregister und 47 Datenregister benötigt werden, ist ein Speichermodul erforderlich.

Der folgende, im NORM-Modus ausgedruckte Streifen zeigt den genauen Ablauf der Ein- und Ausgaben:

```
 XEQ "HAUPT"
MTEIN
A1.1 ?
 4.000      RU
             N
A1.2 ?
 7.000      RU
             N
A1.3 ?
 2.000      RU
             N
A2.1 ?
-6.000      RU
             N
A2.2 ?
 1.000      RU
             N
A2.3 ?
 0.000      RU
             N
A3.1 ?
 2.000      RU
             N
A3.2 ?
 7.000      RU
             N
A3.3 ?
 3.000      RU
             N
```

```
A1.1=4.0000
A1.2=7.0000
A1.3=2.0000

A2.1=-6.0000
A2.2=1.0000
A2.3=0.0000

A3.1=2.0000
A3.2=7.0000
A3.3=3.0000

MTSHF
MTINV

C1.1=0.0600
C2.1=0.3600
C3.1=-0.8200

C1.2=-0.1400
C2.2=0.1600
C3.2=-0.2900

C1.3=-0.0400
C2.3=-0.2400
C3.3=0.9200

MTMUL

E1.1=1.0000
E1.2=0.0000
E1.3=0.0000

E2.1=0.0000
E2.2=1.0000
E2.3=0.0000

E3.1=0.0000
E3.2=0.0000
E3.3=1.0000

MTSPU

SPUR<E>=3.00
```

Hamburg 73, den 01.11.1980 Klaus Werner Hoen
Alaskaweg 18

5.1 MTAIJ

```
LBL"MTAIJ
END
25 BYTES

r=S+2+(i-1)
        *n+j

A:
T:
Z:  j
Y:  i
X:  S(A)

 01•LBL "MTA
IJ"
 02 2
 03 +
 04 X<>Y
 05 INT
 06 1
 07 -
 08 RCL IND
Y
 09 *
 10 +
 11 +
 12 INT
 13 RTN
 14 END
```

5.2 MTEIN

```
LBL"MTEIN
END
88 BYTES

UNTERPROGR.
MTAIJ ERFOR-
DERLICH !

REIHENWEISE
EINGABE DER
m,n-MATRIX A

A:  NAME(A)
T:
Z:  n
Y:  m
X:  S(A)

 01•LBL "MTE
IN"
 02 ASTO IND
X
 03 STO 03
 04 1
 05 +
 06 X<>Y
 07 STO IND
Y
 08 1 E3
 09 /
 10 1
 11 +
 12 STO 06
 13 RDN
 14 1
 15 +
 16 X<>Y
 17 STO IND
Y
 18 1 E3
 19 /
 20 1
 21 +
 22 STO 07
 23•LBL 01
 24 RCL 07
 25 STO 09
 26•LBL 02
 27 CLA
 28 ARCL IND
03
 29 FIX 0
 30 SF 29
 31 RCL 06
 32 INT
 33 ARCL X
 34 CF 29
 35 RCL 09
 36 INT
 37 ARCL X
 38 "├ ?"
 39 X<>Y
 40 RCL 03
 41 XEQ "MTA
IJ"
 42 FIX 3
 43 PROMPT
 44 STO IND
Y
 45 ISG 09
 46 GTO 02
 47 ISG 06
 48 GTO 01
 49 RTN
 50 END
```

5.3 MTAUS

```
LBL"MTAUS
END
100 BYTES

UNTERPROGR.
MTAIJ ERFOR-
DERLICH !

REIHENWEISE
(CF 01) ODER
SPALTENWEI-
SE (SF 01)
AUSGABE DER
m,n-MATRIX A

A:
T:
Z:
Y:
X: S(A)

 01♦LBL "MTA
US"
 02 STO 01
 03 1
 04 +
 05 RCL IND
X
 06 1 E3
 07 /
 08 1
 09 +
 10 X<>Y
 11 1
 12 +
 13 X<>Y
 14 RCL IND
Y
 15 1 E3
 16 /
 17 1
 18 +
 19 FS? 01
 20 X<>Y
 21 STO 07
 22 X<>Y
 23 STO 06
 24 ADV
 25♦LBL 01
 26 RCL 07
 27 STO 09
 28♦LBL 02
 29 CLA
 30 ARCL IND
01
 31 RCL 09
 32 INT
 33 RCL 06
 34 INT
 35 FS? 01
 36 X<>Y
 37 FIX 0
 38 SF 29
 39 ARCL X
 40 CF 29
 41 X<>Y
 42 ARCL X
 43 "⊢="
 44 X<>Y
 45 RCL 01
 46 XEQ "MTA
IJ"
 47 FIX 4
 48 RCL IND
X
 49 RND
 50 ARCL X
 51 AVIEW
 52 FC? 55
 53 STOP
 54 ISG 09
 55 GTO 02
 56 ADV
 57 ISG 06
 58 GTO 01
 59 RTN
 60 END
```

5.4 MT111

```
LBL'MT111
END
68 BYTES

UNTERPROGR.
MTAIJ ERFOR-
DERLICH !

ERZEUGUNG
DER m,m-EIN-
HEITSMATRIX
E

A: NAME(E)
T
Z:
Y: m
X: S(E)

 01+LBL "MT1
11"
 02 ASTO IND
 X
 03 STO 03
 04 1
 05 +
 06 X<>Y
 07 STO IND
Y
 08 X<>Y
 09 1
 10 +
 11 X<>Y
 12 STO IND
Y
 13 1 E3
 14 /
 15 1
 16 +
 17 STO 06
 18 STO 07
 19+LBL 01
 20 RCL 07
 21 STO 09
 22+LBL 02
 23 RCL 09
 24 RCL 06
 25 /
 26 INT
 27 LASTX
 28 /
 29 INT
 30 STO 05
 31 RCL 09
 32 RCL 06
 33 RCL 03
 34 XEQ "MTA
IJ"
 35 RCL 05
 36 STO IND
Y
 37 ISG 09
 38 GTO 02
 39 ISG 06
 40 GTO 01
 41 RTN
 42 END
```

5.5 MTADD

LBL'MTADD
END
108 BYTES

UNTERPROGR.
MTAIJ ERFOR-
DERLICH !

C=A+k*B

S(C)=S(A)
ODER
S(C)=S(B)
ERLAUBT.

WENN A=0,
MUSS S(A)=0
GESETZT WER-
DEN.

A: NAME(C)
T: S(A)
Z: S(B)
Y: k
X: S(C)

```
 01♦LBL "MTA
DD"
 02 ASTO IND
 X
 03 STO 03
 04 RDN
 05 STO 00
 06 RDN
 07 STO 02
 08 RDN
 09 STO 01
 10 X<> T
 11 1
 12 +
 13 X<>Y
 14 1
 15 +
 16 RCL IND
Y
 17 STO IND
Y
 18 1 E3
 19 /
 20 1
 21 +
 22 STO 06
 23 RDN
 24 1
 25 +
 26 X<>Y
 27 1
 28 +
 29 RCL IND
X
 30 STO IND
Z
 31 1 E3
 32 /
 33 1
 34 +
 35 STO 07
 36♦LBL 01
 37 RCL 07
 38 STO 09
 39♦LBL 02
 40 RCL 09
 41 RCL 06
 42 RCL 02
 43 XEQ "MTA
IJ"
 44 RCL IND
X
 45 RCL 00
 46 *
 47 STO 05
 48 RDN
 49 RCL 01
 50 X=0?
 51 GTO 03
 52 RCL Y
 53 +
 54 RCL 02
 55 -
 56 RCL IND
X
 57 ST+ 05
 58 RDN
 59♦LBL 03
 60 RDN
 61 RCL 02
 62 -
 63 RCL 03
 64 +
 65 RCL 05
 66 STO IND
Y
 67 ISG 09
 68 GTO 02
 69 ISG 06
 70 GTO 01
 71 RTN
 72 END
```

5.6 MTSPU

```
LBL'MTSPU
END
49 BYTES
```

UNTERPROGR.
MTAIJ ERFOR-
DERLICH !

Σ = SPUR(A)

```
A:
T:
Z:
Y:  S(A)
X:  REG(Σ)

 01•LBL "MTS
PU"
 02 STO 03
 03 RDN
 04 STO 01
 05 1
 06 +
 07 RCL IND
X
 08 1 E3
 09 /
 10 1
 11 +
 12 STO 06
 13 CLX
 14 STO IND
03
 15•LBL 02
 16 RCL 06
 17 RCL 06
 18 RCL 01
 19 XEQ "MTA
IJ"
 20 RCL IND
X
 21 ST+ IND
03
 22 ISG 06
 23 GTO 02
 24 RTN
 25 END
```

5.7 MTSHF

```
LBL'MTSHF
END
82 BYTES
```

VERSCHIEBEN
DER MATRIX A
VON S(A)
NACH S(B)

```
A:
T:
Z:
Y:  S(A)
X:  S(B)

 01•LBL "MTS
HF"
 02 CF 00
 03 X<>Y
 04 STO 01
 05 -
 06 STO 02
 07 X>0?
 08 SF 00
 09 RCL 01
 10 1
 11 +
 12 RCL IND
X
 13 X<>Y
 14 1
 15 +
 16 X<>Y
 17 RCL IND
Y
 18 *
 19 +
 20 FS? 00
 21 GTO 01
 22 1 E3
 23 /
 24 RCL 01
 25 +
 26 GTO 02
 27•LBL 01
 28 RCL 01
 29 1
 30 -
 31 1 E3
 32 /
 33 +
 34•LBL 02
 35 STO 06
 36•LBL 03
 37 RCL 06
 38 RCL IND
X
 39 X<>Y
 40 RCL 02
 41 +
 42 X<>Y
 43 STO IND
Y
 44 FS? 00
 45 GTO 04
 46 ISG 06
 47 GTO 03
 48 RTN
 49•LBL 04
 50 DSE 06
 51 GTO 03
 52 RTN
 53 END
```

5.8 MTTRA

LBL"MTTRA
END
91 BYTES

UNTERPROGR.
MTAIJ ERFOR-
DERLICH !

B = A'

A: NAME(B)
T:
Z:
Y: S(A)
X: S(B)

```
 01◆LBL "MTT
RA"
 02 STO 03
 03 RDN
 04 STO 01
 05 ASTO IND
 03
 06 1
 07 +
 08 RCL IND
X
 09 RCL 03
 10 2
 11 +
 12 X<>Y
 13 STO IND
Y
 14 1 E3
 15 /
 16 1
 17 +
 18 STO 06
 19 RCL 01
 20 2
 21 +
 22 RCL IND
X
 23 RCL 03
 24 1
 25 +
 26 X<>Y
 27 STO IND
Y
 28 1 E3
 29 /
 30 1
 31 +
 32 STO 07
 33◆LBL 01
 34 RCL 07
 35 STO 09
 36◆LBL 02
 37 RCL 09
 38 RCL 06
 39 RCL 01
 40 XEQ "MTA
IJ"
 41 RCL IND
X
 42 STO 04
 43 RCL 06
 44 RCL 09
 45 RCL 03
 46 XEQ "MTA
IJ"
 47 RCL 04
 48 STO IND
Y
 49 ISG 09
 50 GTO 02
 51 ISG 06
 52 GTO 01
 53 RTN
 54 END
```

5.9 MTMUL

LBL'MTMUL
END
131 BYTES

UNTERPROGR.
MTAIJ ERFOR-
DERLICH !

C = A * B

```
A: NAME(C)
T:
Z: S(A)
Y: S(B)
X: S(C)
```

```
 01•LBL "MTM
UL"
 02 STO 03
 03 RDN
 04 STO 02
 05 RDN
 06 STO 01
 07 ASTO IND
 03
 08 1
 09 +
 10 RCL IND
X
 11 RCL 03
 12 1
 13 +
 14 X<>Y
 15 STO IND
Y
 16 1 E3
 17 /
 18 1
 19 +
 20 STO 06
 21 RDN
 22 RCL 02
 23 2
 24 +
 25 RCL IND
X
 26 1
 27 ST+ T
 28 RDN
 29 STO IND
Z
 30 1 E3
 31 /
 32 1
 33 +
 34 STO 07
 35 RDN
 36 1
 37 -
 38 RCL IND
X
 39 1 E3
 40 /
 41 1
 42 +
 43 STO 08
 44•LBL 01
 45 RCL 07
 46 STO 09
 47•LBL 02
 48 RCL 08
 49 STO 10
 50 0
 51 STO 05
 52•LBL 03
 53 RCL 10
 54 RCL 06
 55 RCL 01
 56 XEQ "MTA
IJ"
 57 RCL IND
X
 58 STO 04
 59 RCL 09
 60 RCL 10
 61 RCL 02
 62 XEQ "MTA
IJ"
 63 RCL IND
X
 64 RCL 04
 65 *
 66 ST+ 05
 67 ISG 10
 68 GTO 03
 69 RCL 09
 70 RCL 06
 71 RCL 03
 72 XEQ "MTA
IJ"
 73 RCL 05
 74 STO IND
Y
 75 ISG 09
 76 GTO 02
 77 ISG 06
 78 GTO 01
 79 RTN
 80 END
```

```
5.10 MTINV

LBL"MTINV
END
189 BYTES

UNTERPROGR.
MTAIJ ERFOR-
DERLICH !

B = A↑(-1)

S(B) = S(A)

A: NAME(B)
T:
Z:
Y:
X: S(A)

 01◆LBL "MTI
NV"
 02 STO 00
 03 ASTO IND
X
 04 1
 05 +
 06 RCL IND
X
 07 STO 01
 08 1 E3
 09 /
 10 1
 11 +
 12 STO 02
 13 STO 03
 14◆LBL 01
 15 1
 16 ENTER↑
 17 1
 18 XEQ 10
 19 RCL IND
X
 20 1/X
 21 STO 06
 22 STO IND
Y
 23 RCL 02
 24 1
 25 +
 26 STO 05
 27◆LBL 02
 28 RCL 05
 29 1
 30 XEQ 10
 31 RCL 06
 32 CHS
 33 ST* IND
Y
 34 ISG 05
 35 GTO 02
 36 RCL 02
 37 1
 38 +
 39 STO 04
 40◆LBL 03
 41 RCL 02
 42 1
 43 +
 44 STO 05
 45◆LBL 04
 46 1
 47 RCL 04
 48 XEQ 10
 49 RCL IND
X
 50 STO 07
 51 RCL 05
 52 1
 53 XEQ 10
 54 RCL IND
X
 55 ST* 07
 56 RCL 05
 57 RCL 04
 58 XEQ 10
 59 RCL 07
 60 ST+ IND
Y
 61 ISG 05
 62 GTO 04
 63 ISG 04
 64 GTO 03
 65 RCL 02
 66 1
 67 +
 68 STO 04
 69◆LBL 05
 70 1
 71 RCL 04
 72 XEQ 10
 73 RCL 06
 74 ST* IND
Y
 75 ISG 04
 76 GTO 05
 77 RCL 02
 78 STO 05
 79◆LBL 06
 80 RCL 05
 81 INT
 82 STO 07
 83 RCL 01
 84 STO 06
 85 XEQ 10
 86 RCL IND
X
 87 STO 08
 88 RCL 02
 89 STO 04
 90◆LBL 07
 91 RCL 06
 92 XEQ 08
 93 STO 06
 94 RCL 07
 95 XEQ 08
 96 STO 07
 97 RCL 06
 98 XEQ 10
 99 RCL IND
X
100 X<> 08
101 STO IND
Y
102 ISG 04
103 GTO 07
104 ISG 05
105 GTO 06
106 ISG 03
107 GTO 01
108 RTN
109◆LBL 08
110 1
111 +
112 RCL 01
113 -
114 X>0?
115 RTN
116 RCL 01
117 +
118 RTN
119◆LBL 10
120 RCL 00
121 XEQ "MTA
IJ"
122 RTN
123 END
```

Hans-Günter Lütke Uphues
Bahnhofstraße 20
4401 Altenberge, 12.11.1980
Tel.: 02505/2149

Steigerung der Rechengeschwindigkeit des HP-41C bei Verwendung des Druckers HP-82143A:

Vielen Benutzern des Taschenrechners HP-41C in Verbindung mit dem Thermodrucker HP-82143 wird es schon aufgefallen sein, daß der 41C bei angeschlossenem und eingeschaltetem Drucker um bis zu 25 % langsamer rechnet (je nach Befehl unterschiedlich; im Mittel ca. 15 %). - Ist der Drucker angeschlossen, aber ausgeschaltet, rechnet der Rechner so schnell wie ohne angeschlossenem Drucker. - Diese Verlangsamung der Rechengeschwindigkeit hat mich sehr gestört, weil ich den Drucker hauptsächlich in Programmen verwende, die eine längere Rechenzeit benötigen.

Im Mai dieses Jahres ist in dem PPC-Calculator-Journal ein Bug-3 - Simulator-Programm von Jan Doig veröffentlicht worden, mit dem jeder HP-41C - Besitzer in die Lage versetzt wurde, den Zustand eines beliebigen Flags zu ändern. Mit diesem Programm habe ich nun bei angeschlossenem und eingeschaltetem Drucker (Flag 55 also automatisch gesetzt) das Druckeranwesenheits-Flag 55 gelöscht. Durch diesen Trick liefen meine Programme so schnell, als wäre der Drucker nicht eingeschaltet, also eine Rechenzeitersparnis von bis zu 20 bis 25 % maximal. Leider mußte ich aber feststellen, daß man die eingebauten Funktionen des Druckers nicht mehr vollständig verwenden kann. Die Befehle ADV, STKPLOT, REGPLOT, PRREG(X), PRBUF und PRΣ können nicht mehr verwendet werden, da sie entweder nicht ausgeführt werden oder der Druck hintereinander erfolgt; es wird kein NONEXISTENT angezeigt.

Andere Befehle wie PRX und PRA können weiterhin verwendet werden, werden aber in einer etwas abgewandelten Form gedruckt. Funktionen (Befehle), die den Buffer des Druckers laden bleiben aktiv, und der Buffer wird auch automatisch zeilenweise ausgedruckt, wenn er "überläuft".

```
01♦LBL "B3"
02 ENTER↑
03 CLA
04 8
05 ST/ Z
06 MOD
07 X<>Y
08 INT
09 1
10 -
11 10↑X
12 RCL d
13 STO \
14 FIX 0
15 CF 29
16 ARCL Y
17 RCL \
18 STO d
19 FC?C IND
          T
20 SF IND T
21 RCL d
22 STO \
23 5
24 LASTX
25 -
26 10↑X
27 FIX 0
28 CF 29
29 ARCL X
30 RCL ]
31 STO d
32 END
```

Glücklicherweise ist es aber dennoch möglich, den Buffer gezielt auf anderem Wege ausdrucken zu lassen: nämlich durch die Befehle PRA,X. Bei diesen beiden Befehlen wird dann ~~beim~~ der Inhalt des X- beziehungsweise Alpha-Registers in den Buffer übernommen, aber noch nicht gleich gedruckt. Erst beim nächsten PRX,A wird der vorherige X- bzw. Alpha-Inhalt gedruckt und der jetzige Wert in den Buffer übernommen. Soweit ist also noch ein PRA, PRX und PRBUF realisierbar. Eine Leerzeile (ADV) läßt sich durch die Befehlsfolge "(space)" PRA realisieren.

Etwas heikel wird die Angelegenheit, wenn man den Buffer durch Befehle wie ACX ~~und~~ ACA und ähnlichen laden möchte: Hierbei werden diese Befehle wohl ausgeführt, aber wenn vorher ein PRX,A - Befehl abgehandelt wurde, steht diese alte Information noch im Buffer, und die neue Information wird hinten angesetzt und beim nächsten Druckvorgang gemeinsam ausgedruckt. Um dies zu verhindern, muß man durch die Befehlsfolge CLA PRA den Buffer ausdrucken, bevor man ihn mit AC-Befehlen neu lädt.

Die VIEW-Befehle führen bei gesetztem Flag 21 zur Programmunterbrechung, wodurch dann das Flag 55, wie auch nach den Befehlen R/S, PROMPT und PSE wieder gesetzt wird. Durch den VIEW-Befehl ist die Information aber in den Buffer geladen und kann somit anschließend ausgedruckt werden.

War ein ADV-Befehl vor dem Löschen des Flags 55 ausgeführt worden, erfolgt der Bufferausdruck <u>immer</u> rechtsbündig. Um linksbündigen Druck zu erreichen, sollte vor dem Löschen des Flags 55 ein PRBUF-Befehl erfolgen.

Insgesamt gesehen ergeben sich einige vielleicht für spezielle Anwendungsfälle unangenehme Einschränkungen, so daß sich die Verwendung dieses beschriebenen Tricks nur für Programme mit langen Rechenzeiten von mindestens mehreren Minuten eignet.

<u>Achtung:</u> Selbstverständlich muß das Löschen des Flags 55 im Programm erfolgen!! (Unterprogrammaufruf von B3 oder durch synthetische Programmierung)

Happy programming . . . . .

Programm zur Ermittlung zweier Quadratzahlen, deren Summe eine Primzahl ist.

$(a^2 + b^2 = PZ)$

Irgend einer der mathematischen Geistes-Heroen (EULER? GAUSS?) fand heraus:

> " Wenn eine Primzahl beim Teilen durch 4 den Rest 1 läßt, dann kann man sie auf genau eine Weise als Summe zweier Quadratzahlen schreiben. "

Einige Beispiele aus einer Tabelle der Primzahlen, ab 12 433 ....

PZ:		a2:		b2:
12 433	=	63^2	+	92^2
12 437	=	71^2	+	86^2
12 451	ERROR			
12 457	=	24^2	+	109^2
12 473	=	32^2	+	107^2
12 479	ERROR			
12 487	ERROR			
12 491	ERROR			

Den Freunden zahlentheoretischer Probleme erleichtert das folgende PRGM die Suche nach den genannten Zahlen.

Dazu noch ein Hinweis zu einer PRGM-Besonderheit: nach meinen fast 10-jährigen Erfahrungen im Umgang mit programmierbaren hp-Taschenrechnern ist die R/S-Taste die am meisten strapazierteste Taste. Es ist daher nur eine Frage der Zeit, daß der "Druckpunkt" dieser Taste erheblich "ermüdet", d.h., mitunter überhaupt nicht mehr spürbar ist. Bei der Arbeit mit dem Rechner irritiert das fast immer. Man sollte diese Taste also weitgehend schonen.

Im hier vorliegenden PRGM wurde daher anstelle von PROMPT nach AVIEW (FS 05) 2-mal PAUSE eingefügt. Während dieser Zeit läßt sich die PRIMZAHL (PZ) bequem eingeben. Die auftretende, kleine Verzögerung dürfte auch für Geschwindigkeitsfanatiker tragbar sein. Wem das nicht gefällt, der kann ja immer noch zu PROMPT und R/S zurückkehren.

PRGM-Ausführung (SIZE mind. = 5)

1. Nach Eingabe der Schrittfolge folgende Neu-Zuordnungen vornehmen:
   - 1.1. LBL Σ ⟶ Taste LN
   - 1.2. LBL DC ⟶ Taste LOG (DA CAIO)
2. START mit LOG; es folgt die Aufforderung, PZ einzugeben. (Beispiel PZ =12497)
3. Anzeige:

   a = 41

   b = 104

   a^2 = 1 681

   b^2 = 10 816

   Σ = 12 497 $(a^2 + b^2)$
4. Wiederholung der Anzeige (beliebig oft) mit LOG

G.SIEWERT (52)

Anmerkung: R/S läßt sich doch jeder beliebigen Taste zuordnen ? Also ......

Oben muß es unter Pkt. 2. heißen:

" START mit LN; es folgt ......"

Ansonsten ist das Prgr. aber völlig O.K., was nicht so selbstverständlich sein mag, da Gerhard Siewert selbst keinen HP-41c besitzt.

Oliver

PRP "Σ"

```
01◆LBL "Σ"
02 CLRG
03 FIX 0
04 "PZ?"
05 AVIEW
06 PSE
07 PSE
08 STO 00
09 4
10 MOD
11 1
12 X=Y?
13 GTO 01
14 0
15 %

16◆LBL 01
17 RCL 00
18 SQRT
19 INT
20 STO 01

21◆LBL 02
22 X↑2
23 RCL 00
24 X<>Y
25 -
26 SQRT
27 STO 02
28 FRC
29 X=0?
30 GTO "DC"
31 RCL 01
32 1
33 -
34 STO 01
35 GTO 02

36◆LBL "DC"
37 RCL 02
38 "a="
39 ARCL X
40 AVIEW
41 PSE
42 PSE
43 X↑2
44 STO 03
45 RCL 01
46 "b="
47 ARCL X
48 AVIEW
49 PSE
50 PSE
51 X↑2
52 STO 04
53 RCL 03
54 "a↑2="
55 ARCL X
56 AVIEW
57 PSE
58 PSE
59 PSE
60 RCL 04
61 "b↑2="
62 ARCL X
63 AVIEW
64 PSE
65 PSE
66 PSE
67 +
68 "Σ="
69 ARCL X
70 AVIEW
71 END
```

Wolfgang Leipold
Kopernikusstr. 6 b

8590 Marktredwitz

Herrn
Oliver Rietschel
Bismarckstr. 31
Postfach 373

2420 Eutin

Lieber Oliver und Mitglieder,

zu dem bisher im Prisma erschienenen Artikel: prima!

Für mich ist der 41 C jedoch mehr Hilfe im Beruf und zur Beschäftigung mit synthetischen Programmen bleibt mir kaum Zeit. Außerdem fehlen mir noch (derzeit) die notwendigen Grundlagen.

Frage: Wieviele im Club sind noch daran interessiert, die Fertigkeit des Programmierens dahingehend auszubauen, Programme für den Beruf, Studium, Schule etc. "zum Laufen zu bringen"? Z.B. sind die Kapitel "Indirekte Operationen" und "Flags" im Handbuch, zumindest für mich, zu kurz gekommen.

Idee: Könnten wir uns nicht ab und zu gegenseitig Aufgaben stellen, wobei die Lösungen allerdings mit kurzem Kommentar versehen werden, damit auch Anfänger und "leicht" Fortgeschrittene erkennen können, "warum" dies oder jenes so gemacht wurde. Die "Aha-Ergebnisse" können vielleicht für den Einzelnen bei weiteren Aufgabenstellungen von Hilfe sein.

So sind z.B. die erschienenen Programme "Addition, Plot 1, Sortieren von Zahlen" etc. nützliche Module, die direkt oder abgeändert verwendet werden können.

Zum "Anstoß" folgende Aufgabenstellung: Es sollen Gehälter ausgezahlt werden an fünf Personen. Im einzelnen beziehen diese Personen folgende Gehälter:

Person	1	DM	4773,--
"	2	DM	3483,--
"	3	DM	2973,--
"	4	DM	2041,--
"	5	DM	1924,--

Es stehen zur Verfügung an Scheinen bzw. Münzen:

20	x	500 DM
30	x	100 DM
30	x	50 DM
30	x	20 DM
10	x	5 DM
15	x	2 DM
14	x	1 DM

Die einzelnen Einheiten sollen so gleichmäßig wie möglich verteilt werden; also Person 1 bekommt nicht etwa acht Scheine a 500 DM, Person 2 sechs Scheine a 500 DM usw., da sonst Person 5 überhaupt keinen 500 DM-Schein mehr bekommen könnte.

Das Programm sollte soweit wie möglich "alleine" laufen, d.h. Dialog, Zwischenspeicherungen etc. sollten minimal sein. Das Ganze unter der Voraussetzung, daß KEIN Drucker zur Verfügung steht.

Macht jemand mit?

Mit den besten Grüßen

Euer

Wolfgang Leipold

FILMLAENGEN-BESTIMMUNG

FUER SUPER-ACHT-FILME

LBL"FILM
END
429 BYTES

SPEICHER:
00 SZENENZAEHLER
01 SUMME BILDER
02 ANZAHL BILDER DER SZ.
03 SUMME METER
04 SUMME MINUTEN
05 BILDER/MINUTE

```
              PRP "FILM"

 01♦LBL "FILM"
5  XEQ 04  FC?C 25
PROMPT  CF 27  CLRG  1
STO 00  CF 01  CF 02
"N BILDER/S:"  PROMPT
ADV  SF 12  CLA  ARCL X
"⊦B"  ACA  SF 13
"ILDER/S"  ACA  PRBUF
CF 12  CF 13  60  *
STO 05

 29♦LBL 00
CF 22  CF 29  FIX 0  ADV
ADV  "SZENE "  ARCL 00
PRA  ADV  ADV

 40♦LBL 05
0  "BILDNUMMER"  PROMPT
X=0?  GTO 01  RCL 01  1
+  X>Y?  GTO 05  RDN
"B"  ACA  SF 13  "ILD "
ACA  CLA  1  ST+ 01  RDN
ARCL 01  ACA  " -"  ACA
ACX  PRBUF  ENTER↑
X<> 01  -  1  +  "SIND"
ACA  ACX  CF 13  " B"
ACA  SF 13  "ILDER"  ACA
PRBUF  STO 02  RCL 05  /
ST+ 04  XEQ 02  RCL 02
.004234  *  FIX 3
ST+ 03  XEQ 03  1
ST+ 00  GTO 00

 96♦LBL 01
ADV  "GESAMTWERTE"  PRA
"==========="  PRA  CLA
ARCL 01  ACA  " B"  ACA
SF 13  "ILDER"  ACA
PRBUF  RCL 04  XEQ 02
RCL 03  XEQ 03  CF 01
CF 02  GTO 99

118♦LBL 03
FIX 3  CF 13  "L"  ACA
SF 13  22  ACCHR  RDN
"NGE"  ACA  ACX  " M"
ACA  PRBUF  XEQ 10
CF 13  RTN

136♦LBL 02
INT  CF 13  "Z"  ACA
SF 13  "EIT"  ACA  ACX
" MIN"  ACA  FIX 1
LASTX  FRC  60  *  ACX
" SEC"  ACA  PRBUF  RTN

157♦LBL 04
FS? 55  GTO 06
"PRINTER"  SF 11  PROMPT
CF 11

164♦LBL 06
ENTER↑  ENTER↑  1  +
X<>Y  "SIZE >= "  FIX 0
ARCL Y  SF 25  RCL IND X
X<>Y  SF 27  RTN

178♦LBL 10
SF 12  "------"  ASTO L
ARCL L  PRA  CF 12  RTN

186♦LBL 99
END
```

GEORG RAABE

----

18 Bilder/s

SZENE 1

Bild 1 - 250
sind 250 Bilder
Zeit 0 min 13.9 sec
Länge 1.059 m

SZENE 2

Bild 251 - 759
sind 509 Bilder
Zeit 0 min 28.3 sec
Länge 2.155 m

SZENE 3

Bild 760 - 854
sind 95 Bilder
Zeit 0 min 5.3 sec
Länge 0.402 m

SZENE 4

Bild 855 - 3600
sind 2746 Bilder
Zeit 2 min 32.6 sec
Länge 11.627 m

SZENE 5

GESAMTWERTE
===========
3600 Bilder
Zeit 3 min 20.0 sec
Länge 15.242 m



Schreiben vom   Blatt Nr. 2

Hewlett-Packard GmbH · Vertriebszentrale · Berner Straße 117, 6000 Frankfurt 56 · Postfach: 560140 · Tel.: (0611) 5004-1 Vermittlung

Oliver Rietschel
Hewlett-Packard Anwender-Club
Postscheckkonto Hmb. 48635-207
Postfach 373 · D-2420 Eutin

Betr.: Batterielebensdauer HP 41C

Sehr geehrter Kunde,

die Batterielebensdauer ist von vielen Faktoren abhängig, wie der Rechner benutzt wird, mit welchen Peripheriegeräten er betrieben wird und wie frisch die Batterie selbst ist.

Wenn Tasten gedrückt werden, oder ein Programm abläuft, befindet sich der Rechner im "RUN" Modus, wenn er nur eingeschaltet ist, im "STANDBY" Modus. Er benötigt dann nur 1/7 der "RUN" Modus Stromaufnahme.

Wird der Rechner nicht benutzt, so schaltet er automatisch aus. Er befindet sich dann im "SLEEP" Modus und benötigt nur etwa 1/1oo der "STANDBY" Stromaufnahme.

Messungen der Batterielebensdauer ergaben die folgenden mittleren Werte:

HP 41C allein	Mittlere Lebensdauer (Stunden)
RUN Modus:	
Stunden bis BAT Indikator	≃ 6o
Stunden bis Anzeige aus	≃ 8o
HP 41C und 82104A Kartenleser	
Kartendurchläufe bis BAT Indikator	≃ 1.ooo
Kartendurchläufe bis nicht mehr gelesen wird	≃ 1.2oo

??????????????????????????Ich denke, wenn BAT-ON, werden keine Karten mehr gelesen ?

Wenn Sie die von Ihnen erzielte Batterielebensdauer mit diesen Werten vergleichen, bedenken Sie folgende Punkte:

1) Die Werte wurden mit frischen Batterien erzielt.

2) Es handelt sich um mittlere Werte, daher können Ihre Werte höher oder niedriger liegen.

3) Kartendurchläuf ist definiert je Seite.

4) Kombinierte HP 41C mit Kartenleser-Verwendung, ergibt niedrigere Werte als die oben angegebenen. Wenn der Kartenlesermotor anläuft, wird die Batterie stark belastet und die Spannung geht zurück.
Gegen Ende der Batterielebensdauer, (jedoch bevor der BAT Indikator erscheint), kann der HP 41C allein noch normal arbeiten, jedoch beim Versuch eine Karte einzulesen, kann die Spannung unter den Referenzwert gezogen werden, sodaß der BAT Indikator erscheint.
Deshalb sind 6o Stunden RUN Betrieb des HP 41C alleine, nicht gleich 1ooo Einlesungen zu setzen.

5) Wenn die Batterien soweit entladen sind, daß keine Karten mehr gelesen werden, sind sie dennoch für den HP 41C Betrieb alleine, noch eine Zeitlang verwendbar. Sie können so Ihre Batterien voll ausnutzen.

6) Speicher oder Anwendermodule verringern die Batterielebensdauer.

7) Wenn ein Drucker 82143A eingesetzt ist, beachten Sie, daß der HP 41C sich während des Druckens im RUN Modus befindet. Längeres Drucken entlädt die Batterie des HP 41C.

Mit freundlichen Grüßen
Hewlett-Packard GmbH
Vertriebszentrale Frankfurt

Kundendienst
Unternehmensbereich
Calculator Produkte

../2

31,32–81

Das sind Strichcodes !

Der Strichcodeleser des HP-41c kann einem nicht nur das Leben erleichtern, indem er schnell Programme einliest und nicht auf dem Tastenfeld vorhandene Funktionen sehr schnell ausführt (z.B. SIZE nach MEMORY LOST), sondern er eröffnet dem Anwender eine Reihe neuer Möglichkeiten. Sehr interessant und ohne Vorkenntnisse anwendbar ist z.B. die nachfolgende Tabelle, die von Jörg Warmuth aus Berlin erstellt wurde. Sie enthält alle 128 ASCII-Codes in codierter Form. Geplottet wurden diese Strichcodes vom Drucker des HP-41c.
Die Vorteile sind enorm: Ohne synthetische Programmierung oder irgendwelche Tricks lassen sich alle ASCII-Zeichen in das Alpharegister eingeben. Dieses ist normalerweise nicht möglich. Beispielsweise findet sich auf der Tastatur kein kleines f mehr: Die Kleinbuchstaben gehen nur bis e .
Damit ist jetzt Schluß: Man nehme seinen Griffel zur Hand, suche das ASCII-Zeichen in untenstehender Tabelle und lese es ein. Bis zu vierundzwanzigmal (danach ist das Alpharegister voll und das erste Zeichen geht wieder verloren). Nicht nur, daß diese Methode viel, viel Speicherplatz spart; die Programme werden wesentlich übersichtlicher. Zeichen wie "&" , Strichmännchen und andere lassen sich sogar in der Anzeige darstellen, die meisten erscheinen jedoch als Null, d.h. als schwarzer Kasten (Black Box). Die Anwendungsbereiche liegen also vornehmlich im Bereich der Druckeranwendung. Mein Vorschlag an HP: Diese Tabelle serienmäßig dem Wand oder dem Drucker beilegen ! Wahrscheinlich ist es sinnvoller, die Tabelle erst dem Drucker beizulegen, denn wer nur den Rechner und den opt. Leser besitzt, hat nur wenig Vorteile und wird sich evtl. über ständige Black Boxes ärgern. Zusammen mit dem Drucker erschließt sie aber eine neue Programmiermöglichkeit, die neben dem Werbeeffekt einen Anreiz bringt, den Lesestift zu kaufen....

Noch ein Wort zu den Strichcodes: Die Tabelle habe ich in teils mühevoller Arbeit auf dem HP-41c mit Thermoprinter erstellt. Die hervorragenden Leseeigenschaften rühren von dem verwendeten schwareen Papier her: erstklassig zu kopieren, dabei absolut wischfest und ohne jeden Lesefehler beim Einlesen mit dem Wand. Ein Bericht über Strichcodeplotten wird sicher noch folgen.

Code	Zeichen	Code	Zeichen	Code	Zeichen	Code	Zeichen
0	◆	32		64	@	96	´
1	×	33	!	65	A	97	a
2	x̄	34	"	66	B	98	b
3	←	35	#	67	C	99	c
4	α	36	$	68	D	100	d
5	β	37	%	69	E	101	e
6	Γ	38	&	70	F	102	f
7	↓	39	'	71	G	103	g
8	Δ	40	(	72	H	104	h
9	σ	41	)	73	I	105	i
10	◆	42	*	74	J	106	j
11	λ	43	+	75	K	107	k
12	µ	44	,	76	L	108	l
13	∡	45	−	77	M	109	m
14	τ	46	.	78	N	110	n
15	Φ	47	/	79	O	111	o
16	Θ	48	0	80	P	112	p
17	Ω	49	1	81	Q	113	q
18	δ	50	2	82	R	114	r
19	Ȧ	51	3	83	S	115	s
20	ȧ	52	4	84	T	116	t
21	Ä	53	5	85	U	117	u
22	ä	54	6	86	V	118	v
23	Ö	55	7	87	W	119	w
24	ö	56	8	88	X	120	x
25	Ü	57	9	89	Y	121	y
26	ü	58	:	90	Z	122	z
27	Æ	59	;	91	[	123	π
28	æ	60	<	92	\	124	|
29	≠	61	=	93	]	125	→
30	£	62	>	94	↑	126	Σ
31	▒	63	?	95	_	127	⊦

Der programmierbare Byte Jumper Rolf Mach (75)

In diesem Artikel möchte ich auf eine, von mir entdeckte, neue Befehlsgruppe eingehen. Ich nenne sie programmierbarer Byte Jumper; kurz PBJ. Es handelt sich dabei um die 2-Byte Befehle 1D, 1E und 1F, die mit o1 bis oF als zweites Byte gekoppelt sind. Laut Hextabelle sind dies GTOa, XEQa und Spare, in Verbindung mit LBL oo bis LBL 14. Im Display erscheinen sie als GTO^T, XEQ^T und W^T. Die Länge des Textes, der in diesen Befehlen steht, hängt vom zweiten Digit des zweiten Bytes ab.

Beispiele: 1D o3 im Programmspeicher ergibt in der Anzeige

GTO^TXYZ

1F o7 ergibt W^TTUVWXYZ

Dabei steht TUVWXYZ für irgendeine Alpha-Kette, da dieser Text ein Abbild der folgenden Bytes ist. Beispielsweise wird hinter GTO^TXYZ LN, X^2 und X geschrieben. Das ergibt dann GTO^TPQR, da PQR in der Hextabelle die zu den Befehlen gehörigen Alphazeichen sind. Um die Haupteigenschaft des PBJ zu verstehen, sollte das folgende Programm eingegeben werden:

```
01 SF 26
02 SF 25
03 GTO "σ"
04 TONE 9
05 STOP
06 BEEP
07 .END.
```

Zeile o3 ist Hex 1D o3 und steht in der Anzeige als GTO^T▩▩▩.

Packen ist notwendig damit keine Null-Bytes (erkennbar an "¯") zwischen PBJ und nächstem Befehl stehen.

Beim Ablauf hört man jedoch nicht TONE 9, sondern nur BEEP. Wird das Programm im SST-Modus durchgegangen, so verharrt der Rechner einen kurzen Moment auf Zeile o3 (er sucht LBL^T▩▩▩, wobei ^{T}▩▩▩ =Hex 9F o9 84; nämlich die Hexcodes für TONE 9 und STOP), um danach mit BEEP fortzufahren. Die drei Bytes TONE 9 und STOP werden also übersprungen. Das SF 25 in o2 ist notwendig, damit das Programm nicht in o3 mit NONEXISTENT anhält.

Hier sieht man die zwei Nachteile des PBJ mit 1D und 1E:

a) Existiert zufällig ein Alphalabel mit dem passendem Text, so wird zu ihm verzweigt.

b) Es muß Flag 25 gesetzt sein.

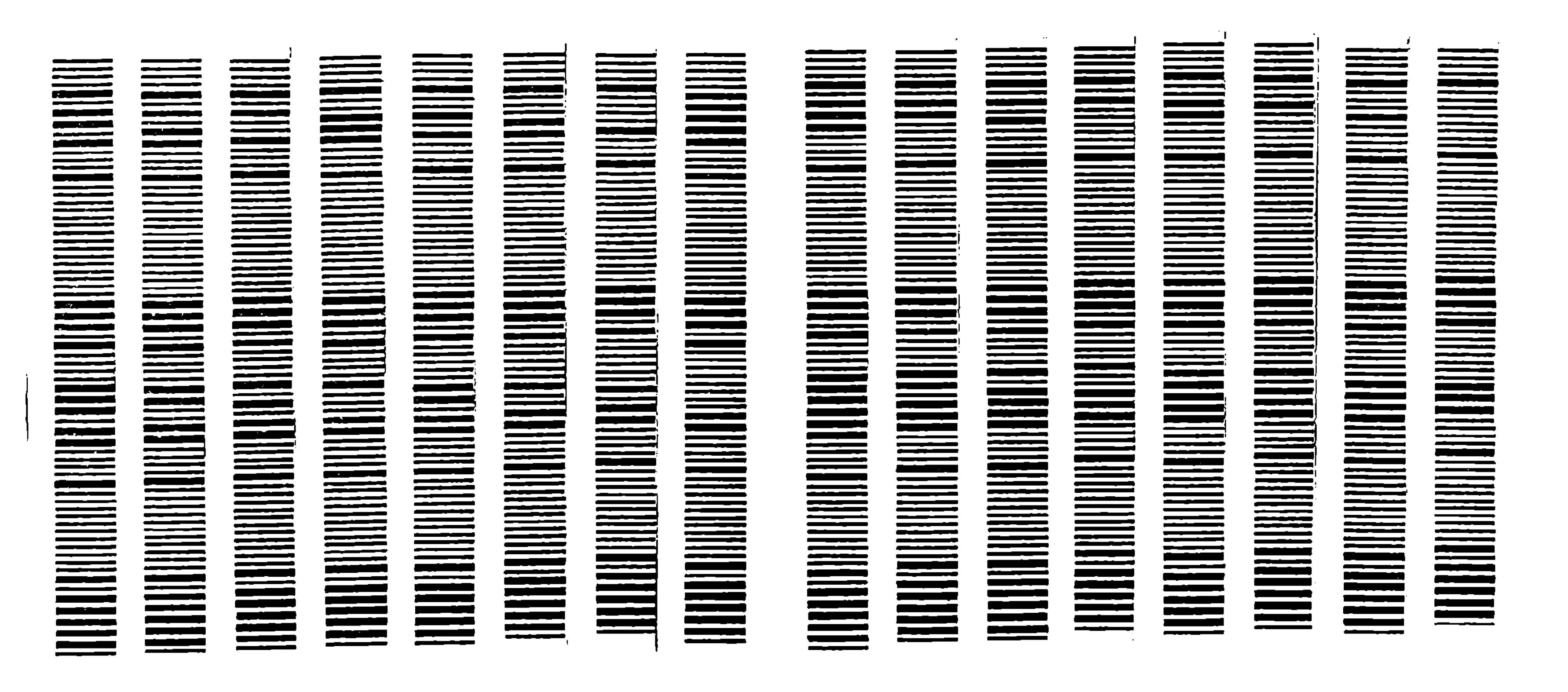
Doch was ist mit 1F?

Dieser PBJ arbeitet nach seltsamen Gesetzmäßigkeiten. Die Auswirkungen reichen von Ausschalten bis Aufhängen, je nachdem welche Peripherie an den HP-41C angeschlossen ist. Bleibt noch das Problem diese Befehle zu erzeugen. Es geht am besten mit Bar Codes oder dem Byte Jumper (STO 29-31, LBL oo-14); KA kann es nicht.

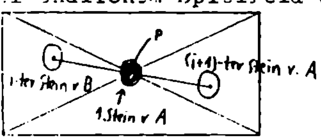
Liebe Mitglieder !

Die Leitung der Abteilung Synthetische Programmierung hat Rolf Mach (75) übernommen. Mit diesem Bericht liefert er seinen "Einstand". Sven Beiersdorf ist die Arbeit zuviel geworden (Was soll ich denn erst sagen ?). Jedenfalls wünsche ich Rolf viel Erfolg und hoffe, daß er eine Einführungsserie in neue synthetische Prgr.techniken zustande bringt. Das bisherige neu zu erläutern erübrigt sich ja: Das Buch William C. Wickes's (siehe Seite 44-81) dürfte da alle Probleme ausräumen. Noch zwei Punkte: Die Bar-Codes machen noch Schwierigkeiten. HP hat sich nicht streng an der Hexcodetabelle orientiert und auch das Buch Bar-Code-Plotting nicht fertig bekommen. Mal sehen ......

Die Tabellen von Matthias Grabiak, die im Weihnachtsinfo "fehlten", konnten nicht erscheinen, weil sie keineswegs abdruckreif waren. Ich hoffe, daß Rolf Mach sie übertragen wird ......



Aufg.1) Gibt es eine Gewinnstrategie, so muß sie auch im Grenzfall $a=b=2r$ zum Erfolg führen. Dann gewinnt A; also muß die Strategie stets A zum Gewinn verhelfen. Im allg. Fall $a,b \geq 2r$ (lt. Regeln gibt es für $a<2r$ oder $b<2r$ keinen Sieger) lautet die Strategie von A:
Der erste Stein ist so zu setzen, daß sein Mittelpunkt mit dem Mittelpunkt P des Spielfelds übereinstimmt.
Seine weiteren Steine setzt er mittelpunktssymmetrisch zum jeweils letzten Stein von B, d.h. A setzt jeweils seinen Stein so, daß die Strecke vom Mittelpunkt seines Steines zu dem des letzten Steines von B durch P halbiert wird.
Da nach Voraussetzung alle Steine denselben Radius haben und keine Steine nach dem Setzen bewegt werden dürfen, kann A nach jedem Zug von B mit dieser "Symmetrie-Strategie" noch einen Zug tun. Bei endlichem Spielfeld wird A also gewinnen.



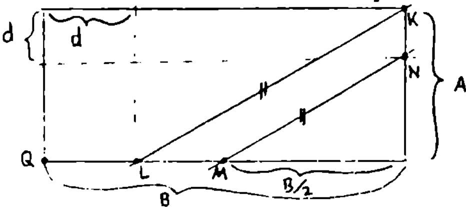
Aufg.2) Das Kreuz besteht aus 2 Balken mit den Flächen Ad und Bd. Die Fläche des Kreuzes, $Ad+Bd-d^2$, soll gleich der Restfläche, also gleich der halben Gesamtfläche sein, d.h.:

$$Ad+Bd-d^2 = \frac{AB}{2} .$$

Die Lösungen dieser quadr.Gl. sind: $d_{1,2} = \frac{A+B \pm \sqrt{A^2+B^2}}{2}$.

d_1 ist zu groß; denn sei o.B.d.A. $0<A=\mathrm{Min}(A,B)$, also $d_1 \geq \frac{2A+\sqrt{2A^2}}{2} \geq A + \frac{A}{\sqrt{2}} > A$. Daraus folgt $d = d_2 = \frac{A+B-\sqrt{A^2+B^2}}{2}$.

Zur Bestimmung von d ist die Lage des Kreuzes unerheblich; man kann die Balken also in eine Ecke parallelverschieben:



Geometrisch läßt sich d bestimmen, indem man 2 Parallelen durch die gegebenen Punkte M und K sucht, sodaß für deren Schnitt=punkte N mit A bzw. L mit B gilt: StreckeQL = StreckeKN (=d). Das ist eine Folgerung aus dem soundsovielten Strahlensatz; denn man erhält aus der ersten o.a. Gl.: $\frac{B-d}{\frac{B}{2}} = \frac{A}{A-d}$.

Werner Stroir
Kampweg 7a
1000 Berlin 27

Berlin-Tegel, 11. 11. 80

Herrn
Oliver Rietschel
HP- Anwender-Club
Postfach 373

2420 Eutin

SUPERGENAU

PLOT F-1 AND F-2
Y: 0.0 TO 1.0
X: 0.0 TO 100.0
STEPS OF 1.000

Sehr geehrter Herr Rietschel!

Programme reizen zum Erproben... und zum Ändern! Oder sollten Sie da etwa anderer Meinung sein?

Mir hatte es das Plot-Programm aus "prisma" 9-80 besonders angetan, da es nach meiner Ansicht besser und übersichtlicher ist, als das PRPLOT. Nun wollte ich schon seit längerer Zeit ein Programm zum gleichzeitigen Plotten von zwei Funktionen schreiben, und Ihre Veröffentlichung war der letzte notwendige Anstoß dazu:

Was daraus geworden ist, können Sie dem beiliegenden Programmausdruck entnehmen! Mit 204 Programmzeilen, die 424 Bytes beanspruchen, ist dieses Programm nur unwesentlich länger als die von Ihnen gelieferte Ausgangsbasis. Die ursprünglich musterhafte Übersichtlichkeit wurde allerdings zum größten Teil geopfert, da ich doch einige Schleifen und andere Tricks einbauen mußte, um Platz zu sparen. Zusammen mit SIZE 026 ergibt sich aber ein Gesamtbedarf von 87 Registern, es ist also mit einem Memory-Modul noch Platz, die zwei zu plottenden Funktionen unterzubringen, zumal diese ja die gleichen Register zur Zwischenspeicherung benutzen können, falls der Stack dazu nicht ausreicht.

Das beiliegende Beispiel zeigt den gemeinsamen Ausdruck von zwei Fourier-Synthesen, die jeweils mit der 1. und 3. bzw. mit der 1. 3. 5. und 7. Harmonischen ein Rechteck anzunähern versuchen. Schnell ist das Programm nicht, die Dauer hängt aber auch stark davon ab, wie umfangreich die Rechengänge in den beiden Funktionen sind. Das gewählte Beispiel dauerte etwa 35 Minuten.

Vielleicht ist dieses Programm eine nette Ergänzung für bereits bekannte Verfahren. Eine Erweiterung auf das gleichzeitige Plotten von 3 Funktionen ist grundsätzlich möglich, nur fürchte ich, daß dann die Angelegenheit doch zu langsam wird.

Als Beispiel früheren "Schaffens füge ich noch zwei Bearbeitungen älterer HP 67/97- Programme bei, die ich für den HP-41C so zurechtgestutzt habe, daß sie jeweils nur eine Magnetkarte benötigen und, vor allen Dingen, mit und ohne angeschlossenen Drucker einwandfrei funktionieren.

Mit besten Grüßen!

Druckprogramm für 2 Funktionen.

```
01•LBL "PLOT"
02 AON
03 "NAME 1?"
04 PROMPT
05 ASTO 05
06 "NAME 2?"
07 PROMPT
08 ASTO 25
09 AOFF
10•LBL "NEU"
11 "Y MIN?"
12 PROMPT
13 STO 00
14 "Y MAX?"
15 PROMPT
16 STO 01
17 "X MIN?"
18 PROMPT
19 STO 02
20 "X MAX?"
21 PROMPT
22 STO 03
23 "X INC?"
24 PROMPT
25 STO 04
26 "PLOT "
27 ARCL 05
28 "├ AND "
29 ARCL 25
30 PRA
31 FIX 1
32 "Y  "
33 ARCL 00
34 "├ TO "
35 ARCL 01
36 PRA
37 "X  "
38 ARCL 02
39 "├ TO "
40 ARCL 03
41 PRA
42 FIX 3
43 "STEPS OF "
44 ARCL 04
45 PRA
46 "-------"
47 ASTO L
48 ARCL L
49 ARCL L
50 ARCL L
51 PRA
52 167
53 RCL 01
54 RCL 00
55 -
56 /
57 STO 08
```

- 01–09: Eingabe der Namen von Funktion 1 und Funktion 2
- 10: Label für Plot mit verändertem Bereich
- 11–13: Untere Grenze für y
- 14–16: Obere Grenze für y
- 17–19: Untere Grenze für x
- 20–22: Obere Grenze für x
- 23–25: Änderung von x
- 26–45: Ausdruck der eingegebenen Namen und Daten
- 46–50: Vorbereitung der y-Achse
- 51: Druck der Achse
- 52–56: Berechnen des Maßstab-Faktors für y
- 57: Faktor speichern

```
58 RCL 02
59 STO 06
60 CF 01
61•LBL 00
62 11.024
63 STO 07
64•LBL 01
65 RCL 06
66 XEQ IND 05
67 CF 00
68•LBL 02
69 RCL 00
70 -
71 RCL 08
72 *
73 1
74 +
75 FIX 0
76 RND
77 168
78 X<>Y
79 X>Y?
80 0
81 ENTER↑
82 0
83 X<=Y?
84 X<>Y
85 STO IND 07
86 FS? 00
87 RTN
88 1
89 ST+ 07
90 RCL 06
91 SF 00
92 XEQ IND 25
93 XEQ 02
94 RCL 04
95 ST+ 06
96 RCL 06
97 RCL 03
98 X<Y?
99 GTO 04
100 ISG 07
101 GTO 01
102 XEQ 05
103 4
104 RCL 04
105 *
106 ST+ 06
107 RCL 06
108 RCL 03
109 X<Y?
110 PRA
111 X<Y?
112 RTN
113 GTO 00
```

- 58–59: x_{min} übernehmen
- 60–63: Schleife für 2x7 y_1/y_2-Werte vorbereiten
- 64–65: x-Wert aufrufen
- 66: $y_1=f_1(x)$
- 67: Kennung f_1
- 68–70: y_{min} abziehen
- 71–72: Anpassung an y-Maßstab
- 73–76: Formatieren, Runden
- 77–84: Prüfung, ob im y-Bereich, evtl. durch 0 ersetzen
- 85: Speichern
- 86–87: Sprung z. 2. Fkt.
- 88–89: Speicheradresse korrigieren
- 90–91: Kennung f_2
- 92: $y_2=f_2(x)$
- 93: y_2 bearbeiten wie y_1
- 94–95: x um Δx ändern
- 96–98: Vergleich mit x_{max}
- 99–101: Rücksprung oder Sortieren
- 102–106: Überspringen des Zeilenabstands, Anfangswert für neue Serie
- 107–109: Vergleich mit x_{max}
- 110: Abschluß-Linie drucken

Stroinski
Nov. 80

Programm	Kommentar
114•LBL 04 115 SF 01 116 0 117 STO IND 07 118 ISG 07 119 GTO 04	Speicher mit $y_1=y_2=0$ laden, wenn x_{max} überschritten
120•LBL 05 121 12.02502 122 STO 10 123 1 124 - 125 STO 09 126 5 E-4	Vorbereiten der Einfügung der Zeilenkennung in Reg. 11...24
127•LBL 03 128 ST+ X 129 ST+ IND 09 130 ST+ IND 10 131 ISG 09 132 ISG 10 133 GTO 03	Additives Einfügen der Zeilenkennung in die Register 11..24
134 12.024 135 STO 10	Sortierschleife vorbereiten
136•LBL 06 137 RCL 10 138 STO 09 139 1 140 - 141 RCL IND X 142•LBL 07 143 RCL IND 09 144 X<Y? 145 X<>Y 146 STO IND 09 147 RDN 148 ISG 09 149 GTO 07	Abfrage der Reg. 11...24, dann 12...24, dann 13... 24 usw. auf kleinsten Wert.
150 STO IND Y 151 ISG 10 152 GTO 06	Abspeichern des kleinsten Wertes
153 11.023 154 STO 09 155 11.024 156 STO 10	Neue Schleife vorbereiten
157•LBL 08 158 ISG 10 159 RCL IND 10 160 INT 161 RCL IND 09 162 INT 163 X≠Y? 164 GTO 09	Prüfung der Reg. auf gleiche ganzzahlige Inhalte
165 LASTX 166 FRC 167 RCL IND 10 168 FRC 169 X<>Y 170 X≠Y?	Aufruf der Zeilenkennung benachbarter Reg. gleichen INT-Wertes.
171 ST+ IND 10 172 CLX 173 STO IND 09 174•LBL 09 175 ISG 09 176 GTO 08	Addition der Zeilenkennung nur, wenn verschieden
177 11.024 178 STO 09 179 0 180 STO 10	Neue Schleife vorbereiten
181•LBL 10 182 RCL IND 09 183 INT 184 X=0? 185 GTO 11 186 ENTER↑ 187 X<> 10 188 - 189 1 190 - 191 SKPCOL	Ermitteln der Kolonnenzahlen und der Sprungbefehle für Druckbuffer
192 RCL IND 09 193 FRC 194 1 E3 195 * 196 ACCOL 197•LBL 11 198 ISG 09 199 GTO 10	Aufruf der Zeilenkennung
200 PRBUF 201 FS?C 01	Ausdruck
202 PRA 203 RTN 204 END	Abschlußlinie

SIZE 026
TOT. REG. 87
424 BYTES.

Zum Problem Zufallzahlengenerator von Erwin Hartmann kann ich sagen, daß die Routine aus der HP Standardprogrammsammlung meines Erachtens schon in Ordnung ist. Die Verschiebung der Häufigkeit nach größeren Zahlen erfolgt durch der Befehl Zeile 46 SQRT in seinem Programm. Dieser Befehl gehört nicht zum Zufallszahlengenerator sondern ist im Programm "Arithmetik-Lehrgang" offensichtlich eingefügt um eine Bevorzugung von schwierigeren Aufgaben mit größeren Zahlenwerten zu erzielen.
Auch das Erscheinen von ∅ ist ganz richtig. Der Generator erzeugt Zufallszahlen von 0–0,999... , wenn 0 als Zufallszahl nicht gewünscht wird, muß eben die anschließende Weiterverarbeitung richtig erfolgen.
Anschließend eine Formel um gleichverteilte Zufallszahlen innerhalb einer maximalen und minimalen Zahl aus obigen Zufallszahlen von 0–1 zu errechnen.

INT(n(max−min+1)+min)

n....Zufallszahl 0–0,999...
max..maximale gewünschte Zahl
min..minimale gewünschte Zahl

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Folgende Programmsammlungen kann ich leihweise zur Verfügung stellen:

Users Library Solutions EE Lab HP67/97
" " " Calendars HP67/97
Statistik-Paket HP67/97
Mathematik-Paket HP67/97

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

In meinem Programm "Startroutine für Zufallszahlengenerator" muß Zeile 03 natürlich richtig heißen: 'OFF, ON'

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Die ACXR-Routine von K.W.Hoenow ist sehr gut, auch ich habe mich mit dem Problem beschäftigt, bin aber auf längere Laufzeiten gekommen. Das Programm funktioniert allerdings nur wenn Flg.29 gelöscht ist. Ich habe das Programm etwas geändert, so das große Zahlen auch mit Trennzeichen richtig untereinander geschrieben werden.

```
01+LBL "ACXR"
02 RND
03 STO Z
04 X=0?
05 GTO 01
06 ABS
07 LOG
08 INT
09 X<=0?
10 GTO 01
11 ST- Y
12 3
13 /
14 INT
15 FS? 29
16 ST- Y
17+LBL 01
18 RDN
19 X<0?
20 CLX
21 SKPCHR
22 RDN
23 ACX
24 END
LBL'ACXR
END        41 BYTES
RY=ANZ. STELLEN VOR KOMM
EINSCHL. VORZ. -2
RX=DRUCKWERT
```

56 Walter Kropf

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Ablehnung: Sammelbestellung Drucker über Tätigk. Nr. 237 : 768 DM + Porto

Automatische E[illegible]ung der möglichen Betriebsdauer bei Verwendung wiederaufladbarer N[illegible]ckel-Kadmium-Akkumulatoren im HP-41 C.

Vorbereitung:

1.) Programm eingeben!
(SIZE 000; TOT. REG.: 7)

2.) Start mit: XEQ/ALPHA/BAT/ALPHA

Anzeige: 4. -02 / 1 / 2 / . / . / . / 38 / 39 — Dauer etwa 60 Sek.

1 MIN. / . / . / . / 2 MIN. / . / usw.! — Hierbei prüfen, ob Wechsel der Minutenanzeige genau nach 60 Sekunden erfolgt, sonst Zeile 07 im Programm entsprechend ändern!

```
01♦LBL "BAT
02 CLST
03 FIX 0
04 " MIN."
05 ASTO Y
06♦LBL 01
07 .039
08♦LBL 02
09 PSE
10 FS? 49
11 OFF
12 ISG X
13 GTO 02
14 RDN
15 1
16 +
17 CLA
18 ARCL X
19 ARCL Y
20 AON
21 GTO 01
22 END
```

Benutzung:

3.) Zu prüfenden Akkumulator einsetzen, dabei Rechner ohne Kartenleser und Drucker betreiben!

4.) Programm starten (XEQ BAT). Rechner läuft, zeigt Betriebsdauer in Minuten an und schaltet sich bei Erschöpfung der Batterie automatisch ab. Die erreichte Betriebsdauer bleibt im ALPHA-Register gespeichert.

5.) Nachdem der Rechner sich selbst ausgeschaltet hat:
Einwandfreie Batterie einsetzen, Rechner einschalten, ALPHA-Modus zeigt erreichte Betriebsdauer an.

Das Wiedereinschalten mit der entladenen Batterie kann zum Verlust der Anzeige und des Programms führen, besonders dann, wenn der Rechner nach der automatischen Abschaltung sehr lange auf seinen Benutzer warten mußte!

Werner Stroinski
18. 12. 80

Wolfgang Butz
Schinnrainstraße 3A
7500 Karlsruhe 41

Alternative zu HP-41C RECHARGABLE BATTERY PAC 82120A 1 Stück 112,-- DM
Zum wechseln------- 2 Stück 224,-- DM

Aufladbare VARTA Lady Batterien	8 Stück a.	12,-- DM =	96,-- DM
Aufnahmebox-Lady (feinelectric)	4 Stück a.	5,-- DM =	20,-- DM
Ladegerät für Mignon Batterien	a.	29,-- DM =	29,-- DM
			145,-- DM

Kauft man statt VARTA/SONY die auch nicht schlecht sein sollen!und hatt man schon ein Ladegerät für Mignon Batterien kann weiteres Geld gespart werden.

Aufladbare Sony-Lady	8 Stück a.	7,-- DM =	56,-- DM
Aufnahmebox-Lady	4 Stück a.	5,-- DM =	20,-- DM
Mignon-Ladegerät ???			29,-- DM
			105,-- DM

Preisbeispiel gekauft im Kaufhaus Karstadt.

mit freundlichen Grüßen
Wolfgang Butz

BÜCHER-ECKE

Liebe Clubmitglieder!

Nach Absprache mit Oliver haben wir uns entschloßen,
an dieser Stelle eine "Bücher-Ecke" einzurichten. Die
Bücher-Ecke soll die Vorstellung, Kritik und ähliches
von Literatur zum Thema "programmierbare"Taschenrech-
ner zum Ziel haben. Dabei werden aber nicht nur Bücher
über HP-Rechner behandelt, sondern -wie erwähnt- über
prgm. Taschenrechner allgemein. Dies hat seinen Grund!
Meistens ist es nämlich recht einfach ein Prgm. für
einen anderen Rechner auf HP's zu "übertragen", bzw.
bei guter Dokumentation einfach auf die angegebenen
Formeln und Methoden zurückzugreifen.
Das sind aber nicht die einzigen Gründe für die Ein-
richtung einer Bücher-Ecke. In meiner nun ca. sechs-
jährigen Erfahrung mit prgm. Taschenrechnern sind mir
-gerade in Hinsicht auf Literatur- nun schon einige
Dinge widerfahren, zum Teil positive, zum Teil nega-
tive, die es notwendig erscheinen ließen, soetwas
einzurichten. Es wäre hier zum Beispiel möglcih, Mit-
glieder von besonders guter Literatur bzw. dem Gegen-
teil in Kenntniss zu setzen und dadurch eventuelle
"Fehlkäufe" zumindest einzuschränken. Man denke hier
auch mal an den Preis dieser Literaturgattung, der
nicht selten um 30,- DM liegt. Und wer würde sich
nicht ärgern, wenn er soviel ausgegeben hätte und
dann feststellen muß, daß das Buch nicht das ist,
was er sich vorgestellt hat oder die darin enthaltenan PRGM.S
überhaupt nicht funktionieren!

Nun noch etwas zur "Organisation" der Bücher-Ecke:
Die Bücher-Ecke soll <u>kein</u> Privileg von Oliver oder
mir sein. Vielmehr sind hiermit a l l e Clubmit-
glieder aufgerufen, sich daran zu beteidigen. Dies
kann durch eigene Artikel, Kritiken, Hinweise/erfahr-
ungen, oder indem Ihr mir Bücher leihweise zusendet,
die ich dann bearbeiten werden, geschehen. Die sorg-
fältige Behandlung meinerseits versteht sich von
selbst und sei hiermit garantiert.
Alles was mit der Bücher-Ecke zusammenhängt bitte
an mich:

Frank Altensen
Uhlandstr. 9
6365 Rosbach III

senden.
Um "Doppeltbearbeitungen" zu vermeiden, seien hier
die Bücher aufgeführt, die mir z. Zt. zur Verfügung
stehen:

H. Alt
Angewandte Mathematik, Finanzmathematik Statistik Informatik für Upn-Rechner
Vieweg

Allgemeine Elektrotechnik Nachrichtentechnik Impulstechnik für Upn-Rechner
Vieweg

Richard Eckert
Die Programmierbaren von HP
Oldenbourg

Horst Kötting
Iteration und Approximation mit Taschenrechnern
Vogel-Verlag

H. Weber/J. Grami
Numerische Verfahren für programmierbare Taschenrechner I
BI Hochschultaschenbuch Bd. 803

Hang Heinrich Gloistehn
Programmieren von Taschenrechnern 1/2/3

H. Schumny
Taschenrechner+Mikrocomputer Jahrbuch 80/81

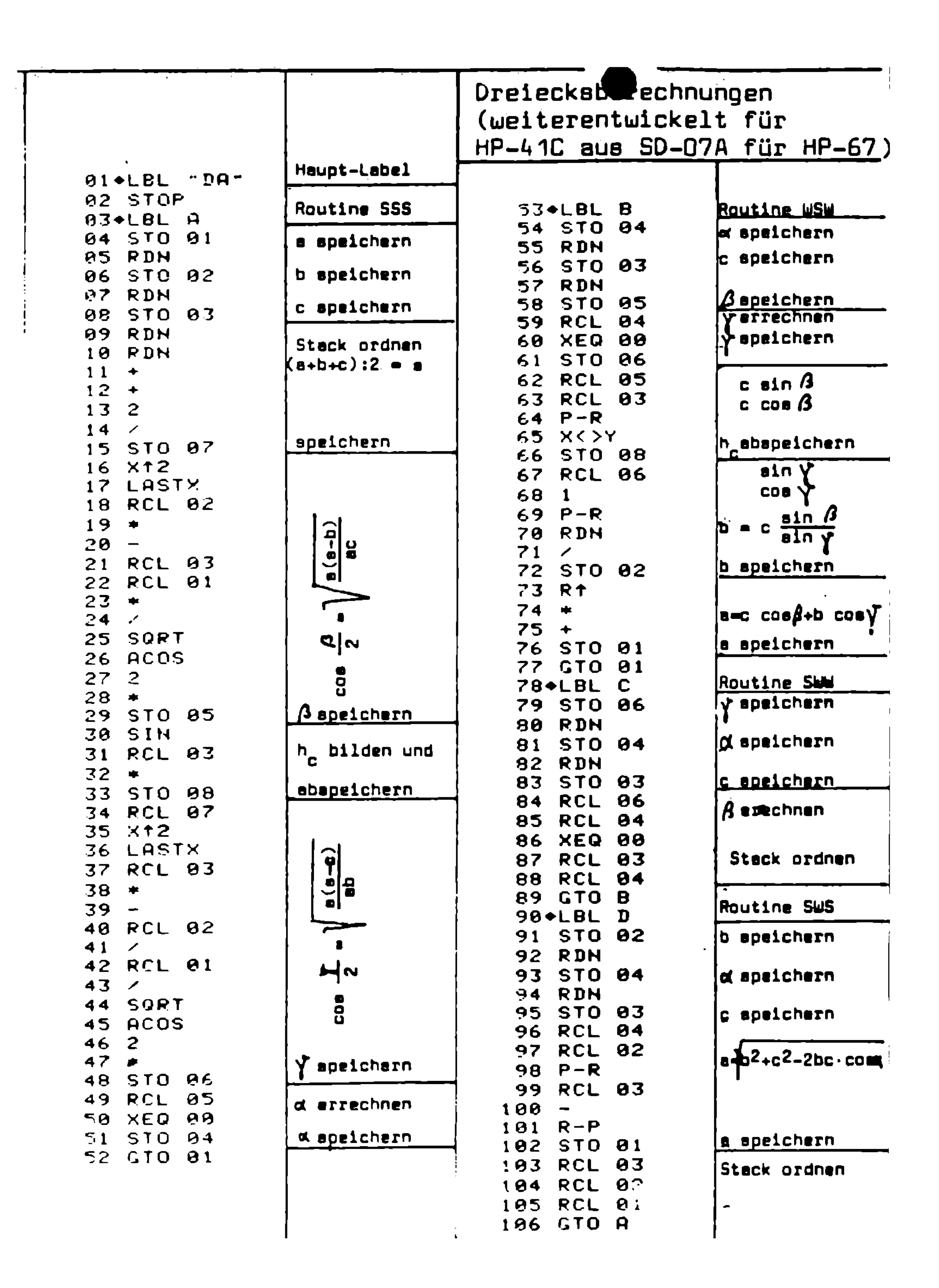
Peter Kahlig
Mathematische Routinen der Physik, Chemie und Technik für AOS-Rechner
Vieweg

Für heute verbleibe ich freundlichst

euer

Frank Altensen

P.S.: Ich wäre euch auch dankbar, wenn ihr mir eure Meinungen oder Vorschläge zur Bücher-Ecke schreiben würdet.



Dreiecksb[illegible]echnungen (weiterentwickelt für HP-41C aus SD-07A für HP-67)

```
01◆LBL "DA"      Haupt-Label
02 STOP
03◆LBL A         Routine SSS
04 STO 01        a speichern
05 RDN
06 STO 02        b speichern
07 RDN
08 STO 03        c speichern
09 RDN
10 RDN           Stack ordnen
11 +             (a+b+c):2 = s
12 +
13 2
14 /
15 STO 07        speichern
16 X↑2
17 LASTX
18 RCL 02
19 *
20 -
21 RCL 03        cos β/2 = √(s(s-b)/ac)
22 RCL 01
23 *
24 /
25 SQRT
26 ACOS
27 2
28 *
29 STO 05        β speichern
30 SIN
31 RCL 03        h_c bilden und
32 *
33 STO 08        abspeichern
34 RCL 07
35 X↑2
36 LASTX
37 RCL 03
38 *
39 -             cos γ/2 = √(s(s-c)/ab)
40 RCL 02
41 /
42 RCL 01
43 /
44 SQRT
45 ACOS
46 2
47 *
48 STO 06        γ speichern
49 RCL 05
50 XEQ 00        α errechnen
51 STO 04        α speichern
52 GTO 01

53◆LBL B         Routine WSW
54 STO 04        α speichern
55 RDN
56 STO 03        c speichern
57 RDN
58 STO 05        β speichern
59 RCL 04
60 XEQ 00        γ errechnen
61 STO 06        γ speichern
62 RCL 05
63 RCL 03        c sin β
64 P-R           c cos β
65 X<>Y
66 STO 08        h_c abspeichern
67 RCL 06
68 1             sin γ
69 P-R           cos γ
70 RDN
71 /             b = c sin β / sin γ
72 STO 02        b speichern
73 R↑
74 *
75 +             a=c cos β+b cos γ
76 STO 01        a speichern
77 GTO 01
78◆LBL C         Routine SWW
79 STO 06        γ speichern
80 RDN
81 STO 04        α speichern
82 RDN
83 STO 03        c speichern
84 RCL 06
85 RCL 04        β errechnen
86 XEQ 00
87 RCL 03        Stack ordnen
88 RCL 04
89 GTO B
90◆LBL D         Routine SWS
91 STO 02        b speichern
92 RDN
93 STO 04        α speichern
94 RDN
95 STO 03        c speichern
96 RCL 04
97 RCL 02        a=√(b²+c²-2bc·cos α)
98 P-R
99 RCL 03
100 -
101 R-P
102 STO 01       a speichern
103 RCL 03       Stack ordnen
104 RCL 02
105 RCL 01
106 GTO A
```

Stroinski
Juli 80

```
107◆LBL E
108 STO 06
109 RDN
110 STO 02
111 RDN
112 STO 03
113 RCL 06
114 SIN
115 RCL 02
116 *
117 RCL 03
118 /
119 ASIN
120 STO 05
121 RCL 06
122 XEQ 00
123 STO 04
124 RCL 05
125 RCL 03
126 RCL 04
127 XEQ B
128 RCL 03
129 RCL 02
130 X<=Y?
131 GTO 02
132 RCL 05
133 COS
134 CHS
135 ACOS
136 STO 05
137 RCL 06
138 XEQ 00
139 STO 04
140 RCL 05
141 RCL 03
142 RCL 04
143 GTO B
144◆LBL 00
145 +
146 COS
147 CHS
148 ACOS
149 RTN
```

Routine SSW
- γ speichern
- b speichern
- c speichern
- $\sin\beta = \sin\gamma \cdot \frac{b}{c}$
- β speichern
- α errechnen
- α speichern
- Stack ordnen
- 1. Lösung
- 2. Lösung möglich?
- $\beta' = \gamma - \beta$ errechnen
- β' speichern
- α' errechnen
- α' speichern
- Stack ordnen für 2. Lösung
- Unterprogramm für 3. Winkel

Registerinhalte:
- 00 Schleifensteuerung
- 01 a
- 02 b
- 03 c
- 04 α
- 05 β
- 06 γ
- 07 Fläche
- 08 Höhe h_c

```
150◆LBL 01
151 RCL 08
152 RCL 01
153 *
154 2
155 /
156 STO 07
157 1,007
158 STO 00
159◆LBL 03
160 RCL IND 00
161 FS? 55
162 PRX
163 FS? 00
164 STOP
165 PSE
166 PSE
167 RCL 00
168 3
169 MOD
170 INT
171 X=0?
172 ADV
173 ISG 00
174 GTO 03
175◆LBL 02
176 ADV
177 RDN
178 RDN
179 RTN
180 STOP
181 END
```

SIZE 009
TOT. REG. 41
USER MODUS
A,B,C,D,E.

DAUERANZEIGE ABRUFBAR:
FLAG 00 SETZEN. MIT R/S STEUERN.

Anzeige- und Druckroutine
- Dreiecksfläche errechnen und speichern
- Schleife vorbereiten
- Leerzeile nach 3. Wert

Der Ausdruck bzw. die Anzeige erfolgt in der Reihenfolge:

a
b
c
α
β
γ
Fläche
(danach eventuell die 2. Lösung in der gleichen Reihenfolge)

Der automatische Ablauf der Anzeige kann durch Betätigen der R/S-Taste jederzeit gestoppt und später (durch erneutes Betätigen der R/S-Taste) wieder freigegeben werden.
Vorheriges Setzen des Flags 00 führt zu automatischem Stop bei jedem Wert. Abruf des nächsten Wertes wiederum durch die R/S-Taste.

Testbericht externe Programm- und Datenspeicher HP82104A und HP82153A

HP82104A - Magnetkartenleser

Klein und handlich ist er schon - der Kartenleser 82104A. Seit Anfang an gehört er mit zum System 41. Er wird einfach auf den "Rücken" des HP-41c(v) aufgesetzt, wobei ein "Beinchen" in Port 4 hineinreicht und so die Verbindung zum Systembus herstellt. Dabei werden die Ports 1 und 2 verdeckt - sie können also "nur" für Module verwandt werden. Port 3 bleibt für eine weitere Peripherieeinheit frei- sei es nun ein Drucker oder der Strichcodeleser- oder noch ein RAM/ROM. Damit ist es nicht möglich, Drucker und Strichcodeleser gleichzeitig am Rechner angeschlossen zu haben. Sicherlich nicht erfreulich, aber baulich nun einmal unabdingbar. Bleiben wir gleich einmal bei der Mechanik: Seit einem Jahr habe ich zwei Kartenleser aus gleicher Fertigungsreihe und Bestellung laufen (2011S45049). Reklamationen gab es keine, doch traten einige "Besonderheiten" auf. Aus dem direkten Vergleich der beiden Testmodelle und versch. Gesprächen ließ sich unschwer erkennen, daß Kartenleser mehr oder weniger Einzelprodukte sind. Das macht bereits das Motorengeräusch deutlich: Der eine summt wie eine Mücke, der andere brummt wie eine dicke Hummel im Sturzflug. Dennoch kommt nicht die Vermutung auf, es könnte darin eine kleine Motorsäge versteckt sein, wie es in einem Test des TI-59 formuliert wurde. Mechanisch darf man also schon einen Pluspunkt verteilen: leise, handlich, formschön. Auch mit den Steckkontakten gab es keinerlei Schwierigkeiten und die Stromversorgung ist ebenfalls zufriedenstellend gelöst: Erst die eingeschobene Magnetkarte schaltet die Stromversorgung für den Motor ein. Mit grob gerundeten Verbrauchswerten von 150/200 mA für Lesen/Schreiben ist er zwar das durstigste Teil des Systems, das durch den Rechner versorgt wird, damit aber immer noch deutlich begnügsamer als der TI-59, der ein Vielfaches dessen braucht. Ernstzunehmende mechanische Beanstandungen gibt es beim fabrikneuen Gerät also keine und das - man sollte es nicht verschweigen - hat mit ca. 450,--DM ja auch seinen Preis.
Kommen wir aber zum nächstem Testpunkt - der Haltbarkeit. Hier mit dem TI-59 zu vergleichen, wäre blanker Hohn - HP hat noch nie versucht, der berühmt berüchtigten TI-Qualität nachzueifern. Um es vorwegzunehmen: Einen Garantiefall gab es nicht und die Geräte machten einen insgesamt durchaus befriedigenden Eindruck. Nennen wir das eine, vielbenutzte Gerät A und das andere B, so läßt sich folgendes Resumee ziehen: Während A durchschnittlich etwa 20 Karten pro Tag verarbeiten durfte, waren es bei B ca. 3. Der relativ hohe Wert bei A ist durch etwa 10 bis 20 Karten alle 3 bis 4 Tage und in "Stoßzeiten" dann wiederum bis zu 400 Karten in bis zu zweieinhalb Stunden bedingt. B wurde dagegen ziemlich gleichverteilt über alle Tage mit Karten gefüttert. Von den Reinigungskarten wurde sinnvoller Weise nur mäßiger Gebrauch gemacht, da diese jedesmal ein klein wenig von dem Schreib-Lese-Kopf abhobeln. Wie sieht es nun heute aus ? A ist die o.g. Mücke; B die Hummel. Möchte man mit B mehrere Karten beschreiben, so treten vereinzelt Schreibfehler auf - begleitet von einem leisen aber unüberhörbaren Quietschen. A hat entgegengesetzte Macken: Er ist in die Meldungen CHECKSUM ERROR und MAL-FUNCTION verliebt, was soviel wie Lesefehler und/oder verdreckte Karte bedeutet. Die "Ausfallquote" beträgt ca. 5 bis 10% der Karten. Abhilfe: Noch einmal beschreiben oder lesen - meistens klappt es. Sonst hilft einfaches Abreiben mit einem sauberen Tuch - nötigenfalls vorher in reinem Alkohol getränkt (feucht, nicht triefend). Extrem selten kommt es vor, daß eine Karte nur gut zum wegwerfen ist. Gerade wo ich diese Zeilen schreibe, bekomme ich vom Walter Pieperhoff noch einen Tip: Statt Reinigungskarte lieber ab und zu mal kräftig durch den Kartenschlitz blasen (nicht sebern !!). Das wirkt praktisch genausogut, ist aber wesentlich schonender für den Kopf. Ich habe dies gleich ausprobiert: siehe da, es stimmt!!!

Der Kartenleser läßt sich sehr vielseitig einsetzen. Neben div. Funktionen zur Prgr. und Datenspeicherung bringt er alle Befehle des HP-67/97 mit. In Verbindung mit dem ebenfalls im Kartenleser enthaltenen Language-Compiler oder besser Translator, der vom HP-67/97 beschriebene Karten in den Maschinencode des 41c(v) übersetzt, werden 67/97 und 41 aufwärtskompatibel. Das Übersetzen wird notwendig, weil die Hexcode-Tabelle des 67/97 anders aufgebaut ist als die des 41 (prisma 27/1980)
Für die Aufzeichnungen auf Magnetkarte gibt es prinzipiell vier versch. Arten:

<u>Programmkarten:</u> Wird eine Magnetkarte im PRGM-Modus eingeschoben, so wird das gerade im RAM befindliche Programm bis zum nächsten END oder .END. hin aufgezeichnet.
<u>Datenkarten:</u> Mithilfe der Befehle WDTA und WDTAX lassen sich alle oder ein Teil (definierbar-softwaremäßig) Datenregister (numerisch bezeichneten, also o1,o2,o3,....) auf Magnetkarte speichern und mit RDTA wieder einlesen. Das gezielte einlesen in bestimmte Datenregister ist mit RDTAX möglich.(X enthält aaa.eee ,wobei aaa-Anfangs- und eee-Endadresse)
<u>Statuskarten:</u> Mit WSTS lassen sich die 16 Statusregister (siehe Bericht Rechnerorganisation) aufzeichnen. Außerdem werden alle Tastenbelegungen abgespeichert.
<u>Speicherauszugskarten:</u> Auf Speicherauszugskarten sind alle RAM-Daten des Rechners geschrieben. Mit WALL werden die Daten aufgezeichnet, die bei Programm-,Daten- und Statuskarten zusammen abgespeichert werden. Nach MEMORY-LOST (Master Clear) und Einlesen eines WALL-Kartensatzes befindet sich der Rechner in exakt dem gleichen Zustand wie beim Beschreiben dieser Karten.
Programm-,Status- und Speicherauszugskarten kann man jederzeit im Normal-Modus einlesen. Durch Abschneiden der Ecken einer Karte kann man die Karten vor unbeabsichtigtem Löschen oder Überschreiben schützen. Mit SF 14 kann man diesen Eckenabschnitt jedoch in seiner Wirkung eliminieren. Interessante und nützliche Befehle sind RSUB (Einlesen eines Unterprogrammes) und MRG (Anhängen eines Programmes an ein anderes), die viele Möglichkeiten eröffnen. Mit WPRV lassen sich Programme auf Magnetkarte schreiben. Diese Programme können dann zwar noch eingelesen und abgearbeitet werden, aber das Kopieren, Ansehen oder Auflisten ist nicht mehr möglich (fast nicht...) Darüber hinaus verfügt der Rechner noch über ein paar andere Befehle, wenn der Kartenleser angeschlossen ist, die dessen Bedienung erfreulich erleichtern und teils sehr nützlich sind. Zusammenfassend läßt sich also sagen, daß der Kartenleser sein Geld wert ist, wenngleich die Methode der Datenspeicherung auf Magnetkarten keineswegs mehr zur modernen Kategorie zählt. Von einem Kartenleser läßt sich aber nicht mehr erwarten - der HP821o4 könnte der Höhepunkt der Kartenleser-Ära sein. Meine Bitte um Genehmigung des Abdrucks der Kurzanleitung vom 2.2.81 wurde bisher von HP leider nicht beantwortet.

<u>HP82153A - Barcode-Leser</u>

Ob nun Barcode- oder Strichcode-Leser, ob WAND oder Griffel: Gemeint ist immer der HP82153A. Er ist keine Alternative zum Kartenleser, sondern eine Ergänzung für andere Anwendungszwecke. Sein Nutzen ist stark umstritten.Hauptargument: Man kann seine Strichcodes nicht selbst plotten. Wie war ! Zumal HP es noch nicht einmal geschafft hat, eine Anleitung, wie man sich Codes plotten kann, herauszubringen und auch auf meine Anfrage, wann mit dem Buch zu rechnen sei, nicht antwortet, ist die vordringliche Frage, was man mit dem Lesestift denn anfangen soll. Zwar verfügt der Lesestift über eine Reihe von Funktionen und Befehlen, doch mag das alles nicht über den effektiv noch fehlenden Nutzen hinwegtäuschen. Solange man auf Großrechner angewiesen ist wird ein Nutzen auf breiter Front nicht möglich sein. Daher soll ein ausführlicher Testbericht erst dann verfaßt werden, wenn es möglich ist, auch mit dem 41c Strichcodes zu plotten. Ob es auf der Hannovermesse die Lösung des Problems gibt ? Wir werden sehen !

<u>Wofür Strichcodes ?</u> Abspeicherung großer Datenmengen, die garnicht verändert oder korrigiert werden (z.B. viele lange Programme). Billiger als Magnetkarten, einfacher zu verbreiten. Ideal für Zeitschriften u.ä.
<u>Wofür Magnetkarten ?</u> Täglicher Gebrauch. Entwurf,Entwicklung,Testen und Verbessern von Programmen. Abspeichern von sich ändernden Datenmengen (z.B. Zahlenwerte) Alles, was mal so eben an abzuspeichernden Sachen anfällt.

Raimund Berg, Hagener Straße 2oo, 591o Kreuztal-Eichen

Das Programm dient zur monatlichen Lohn- und Gehaltsabrechnung. Es errechnet den Bruttolohn, die steuer- und sozialversicherungspflichtigen Beträge, die Abzüge einschließlich VWL und Sozialversicherung und bestimmt den Auszahlungsbetrag. Mit dem Programm kann in einfacher Weise die monatliche Lohn/Gehaltsabrechnung erstellt oder überprüft werden.

Ausführung: Nach jeder Dateneingabe und jedem PROMPT Rechner mit R/S starten. Bei der Dateneingabe dürfen 2 Stackregister benutzt werden. Ist eine Eingabe nicht erforderlich, nur R/S drücken.

Schritt:	Tasten:	Anzeige:	Eingaben:
1	XEQ'LST	ST.KL?	Steuerklasse 1 bis 6 eingeben
2	R/S	KINDER?	Kinderanzahl laut Lohnsteuerkarte eingeben
3	R/S	RELIGION?	Religionsgemeinschaft laut Lohnsteuerkarte eingeben: E = ev, K = rk, Space = ohne 1. Buchstabe = Steuerpflichtiger 2. Buchstabe = Ehegatte Beispiel: KK = beide Ehegatten rk E_ = Steuerpflichtiger ev, Ehegatte ohne Religionsgemeinschaft Rechner hält im ALPHA-Modus an.
4	R/S	ALTER?	Lebensalter eingeben. Ab 63 Jahre keine Arbeitslosenversicherungsbeiträge, ab 62 Jahre Altersentlastungsbetrag
5	R/S	GEHALT?	Bei Angestellten monatliches Grundgehalt eingeben. Weiter bei Schritt 15. Wenn Arbeiter, weiter bei Schritt 6.
6	R/S	ECKLOHN?	Basisstundenlohn eingeben
7	R/S	GSTD.=?	Gesamte Arbeitsstunden einschl. Überstunden eingeben
8	R/S	GSTD.=...	Betrag der Gesamtstunden
9	R/S	Z-STD=?	zusätzliche Stunden mit Zuschlag eingeben (z. B. Überstunden, Nachtstunden, Sonntagsstunden, Stunden mit Schmutzzulage usw.) Hier wird nur die zusätzliche Vergütung berechnet. Der Grundlohn ist bei Schritt 7 mit berücksichtigt worden. *Ohne Eingabe, weiter bei 15.*
1o	R/S	% ?	Zulage in % des Ecklohns. Wenn keine Eingabe, weiter bei Schritt 15.
11	R/S	%=...	Betrag der Zulage pro Stunde
12	R/S	Z-STD=...	Betrag der Zulage gesamt
13	R/S	STPFL=?	Zulage Steuerpflichtig? (Nacht-, Sonntags-, Feiertags-, usw.-zulagen sind steuer- und sozialversicherungsfrei.) Wenn ja, 'J eingeben, Rechner hält im ALPHA-Modus an.
14	R/S	SOZV.=?	Zulage sozialversicherungspflichtig? Wenn ja, 'J eingeben, Rechner hält im ALPHA-Modus an. Rechner geht nach Schritt 9.
15	R/S	VWL AN/AG?	Vermögenswirksame Leistungen: Anteil Arbeitnehmer, ENTER↑, Anteil Arbeitgeber. Ist ein Anteil gleich Null, so muß diese mit eingeben werden.

Schritt:	Tasten:	Anzeige:	Eingaben:
16	R/S	GRATI.?	Ggf. Weihnachtsgratifikation eingeben. Bei Dateneingabe wird 6oo,- DM Steuerfreibetrag und bis zu 1oo,- DM Sozialversicherungsfreibetrag berechnet.
17	R/S	SONSTIGES?	Sonstige Bezüge, die zum Bruttolohn gehören, eingeben. (z. B. zusätzliches Urlaubsgeld, Lohnfortzahlung, usw.) Rechner fragt anschließend, ob der Betrag sozial- bzw. steuerpflichtig ist. Ist der Betrag nur teilweise sozialversicherungs- oder steuerpflichtig, müssen die Teilbeträge getrennt eingegeben werden. Wenn keine Eingabe, weiter bei Schritt 2o.
18	R/S	STPFL=?	Betrag steuerpflichtig? Wenn ja, 'J eingeben, Rechner hält im ALPHA-Modus an.
19	R/S	SOZV.=?	Betrag sozialversicherungspflichtig? Wenn ja, 'J eingeben, Rechner hält im ALPHA-Modus an. Weiter bei Schritt 17.
2o	R/S	FREIBETRAG?	Steuerfreibetrag laut Lohnsteuerkarte eingeben. (Nicht den Weihnachtsfreibetrag, Altersentlastungbetrag, steuerfreie Bezüge)
21	R/S	STPFL=...	steuerpflichtiger Betrag
22	R/S	SOZV.=...	sozialversicherungspflichtiger Betrag
23	R/S	BRUTTO=...	Bruttolohn bzw. -gehalt
24	R/S	LST=...	Lohnsteuerbetrag
25	R/S	E KST=...	Wenn ev. Kirchensteuerpflicht, erfolgt Anzeige.
26	R/S	K KST=...	Wenn rk. Kirchensteuerpflicht, erfolgt Anzeige.
27	R/S	G=...	Krankenversicherungarbeitnehmeranteil. Das Programm berücksichtigt bei den Sozialversicherungen die Beitragsbemessensgrenze sowie die Geringverdienergrenze. Ab der Beitragsbemessensgrenze sind Angestellte beitragsfrei.
28	R/S	K=... oder L=...	Arbeitnehmeranteil Rentenversicherung Arbeiter " " Angestellte
29	R/S	M=...	Arbeitnehmeranteil Arbeitslosenversicherung. Ab 63 Jahre beitragsfrei.
3o	R/S	VWL=...	VWL-Gesamtabzug
31	R/S	ABZUEGE=?	Abzüge für Abschlagszahlungen, Lohnaufrechnung, usw.
32	R/S	ABZUEGE=...	Gesamtabzüge
33	R/S	NETTO=...	Nettolohn
34	R/S	VRS.K=181,5o	Arbeitgeberanteil der Krankenversicherung für freiwillig versicherte Angestellte mit einem Gehalt über der Beitragsbemessensgrenze.
35	R/S	SPARZUL.=...	Sparzulage für VWL-Beträge
36	R/S	ERSTATTUNG ?	Erstattungen für Fahrgelder, Auslagen usw. eingeben
37	R/S	AUSZAHLEN=...	Auszahlungsbetrag

Abzüge vom Bruttolohn: Schritte 24,25,26,27,28,29,3o,31
Hinzurechnungen: Schritte 34,35,36

Das Programm überprüft nicht alle Daten vollständig auf Richtigkeit. Somit kann bei falscher Eingabe u. U. eine fehlerhafte Berechnung erfolgen. Zeigt das Programm DATA ERROR an, empfiehlt es sich, mit XEQ'LST neu zu beginnen.

Das Programm berücksichtigt VWL bis 52 DM/Monat. Der darüber gehende Betrag erhält keine Sparzulage und muß im Rahmen der Abzüge (Schritt 31) eingegeben werden.

Das Programm behandelt jeden Angestellten mit einem Bruttogehalt über der Beitragsbemessensgrenze als freiwillig krankenversichert. Ist dies nicht der Fall, so müssen die 181,5o DM Arbeitgeberanteil an der Krankenversicherung nicht hinzügerechnet werden und 181,5o DM Arbeitnehmeranteil dem Nettolohn abgezogen werden. Der Beitrag von 181,5o DM gilt natürlich nur bei einem Beitragssatz von 11,o % bzw. 5,5 % für jeweils Arbeitgeber und Arbeitnehmer.

Die Erstattungs- und Abzugsbeträge müssen jeweils in einer Summe eingegeben werden.

Das Programm muß ggf. den jeweiligen Erfordernissen angepaßt werden, da die Lohnabrechnung ziemlich umfangreich sein kann und die Konstanten sich recht schnell ändern können. Das abgedruckte Programm benutzt folgende einprogrammierte Werte:

Zeile:	Bemerkung:
146	Höchstbetrag der zulagefähigen VWL
172	Altersentlastungsbetrag 25o DM/Monat höchstens
193	Vorsorgepauschale 9 % vom Jahresgehalt
195	Vorsorgepauschale höchstens jedoch 9 % der Jahresrentenbeitragsbemessensgrenze
2o2	Weihnachtsfreibetrag 6oo DM pauschal
2o4,2o5	Weihnachtsfreibetrag der Sozialversicherung bis zu 1oo DM
24o-249	Rundung auf die jeweiligen Eingangsstufen des Monatstarif, ab 1981 4,5o DM-Stufen
254	Summe der Jahresfreibeträge Steuerklasse I und IV ohne Vorsorgepauschale
26o	zusätzlicher Jahresfreibetrag bei der Steuerklasse II, wenn Kinder zu berücksichtigen
262	Grundjahresfreibeträge Steuerklasse II
267	Summe der Jahresfreibeträge Steuerklasse III ohne Vorsorgepauschale
293	Summe der Jahresfreibeträge Steuerklasse V
295	Rundungsbetrag Steuerklasse VI
329-338	Berechnung des Kinderfreibetrags bei der Kirchensteuer
344	Kirchensteuersatz 8 %, in Berlin, Hessen, Niedersachsen, Nordrhein-Westfalen, Rheinland-Pfalz, Saarland und Schleswig-Holstein 9 %.
349-351	Mindestbetrag der Kirchensteuer, nur in Baden-Württemberg, Hamburg, Niedersachsen, Schleswig-Holstein o,6o DM, in Hessen o,3o DM, in den anderen Ländern kein Mindestbetrag
371	die Hälfte der Monatslohnsteuer-Eingangsstufen, ab 1981 2,25 DM
374	1o % der Beitragsbemessensgrenze, 1981 44o DM. Achtung: dieser Betrag ändert sich jedes Jahr.
389	Arbeitnehmeranteil an der Krankenversicherung, bei jeder Krankenkasse unterschiedlich, ∅ ca. 5,5 %
4o1	Arbeitnehmer anteil an der Rentenversicherung, ab 1981 9,25 %
413	Arbeitnehmeranteil an der Arbeitslosenversicherung, 1,5 %
445	Höchstbetrag des Krankenversicherungsbeitrags: 1981: 5,5 % von (75 % der Beitragsbemessensgrenze (44oo,- DM))

Zeile:	Bemerkung:
461	,3 = 3o % Sparzulage bei VWL bis einschl. 2 Kinder.
463	,4 = 4o % Sparzulage bei VWL ab einschl .3 Kinder.
477	Betrag zur Berechnung der Vorsorgepauschale, ab 1982 468o DM
482	Betrag zur Berechnung der Vorsorgepauschale
5o6, 5o9	Rundungsbetrag der Jahreslohnsteuer, ergibt die Eingangsstufen der Jahressteuer
523-61o	Berechnung der Jahressteuer nach § 32a ESTG

Abänderung des Programms für 198o, 1981, 1982 (abgedruckt 1981)

Zeile:	198o:	1981:	1982:
195	4536	4752	9 % der Jahresbeitragsbemessensgrenze
24o	,5	3	3
242 243	,4 x	4,5 /	4,5 /
245 246 247	,4 / 1,99	4,5 x 1,49	4,5 x 1,49
254	1794	1314	1314
26o	216o	2136	3348
262	2634	2178	2178
267	2544	1584	1584
295	24	18	18
371	1,25	2,25	2,25
374	42o	44o	1o % der Monatsbeitragsbemessensgrenze
445	173,25	181,5	Höchstbeitrag zur Krankenversicherung bei 5,5 %
477	42oo	42oo	468o
5o5-51o	LBL 18 48E3 X<>Y X<Y? SF o7 6o FS?C o7 XEQ 17 / LASTX X<>Y INT x RTN	LBL 18 54 / INT 54 x RTN	LBL 18 54 / INT 54 x RTN
529	4,8	6	6
533	1,6	1,8	1,8
536	3,2	4,2	4,2
542	16o19	18E3	18E3

Zeile:	198o:	1981:	1982:
547	1o,86	3,o5	3,o5
549	154,42	73,76	73,76
553	925	695	695
561	27o8	3o34	3o34
566	13oo19	13E4	13E4
571	,1	,o9	,o9
573	6,o7	5,45	5,45
577	1o9,95	88,13	88,13
581	48oo	5o4o	5o4o
585	15298	2oo18	2oo18
593	13664	14837	14837
598	3719	4213	4213
6o7	812	926	926

Bedeutung der Flags, wenn sie gesetzt sind:

Foo: Angestellter
Fo1: Steuerklasse I
Fo2: " II
Fo3: " III
Fo4: " IV
Fo5: " V
Fo6: " VI
Fo7: benutzt
Fo8: "

F15: rk. Kirchensteuerpflicht
F16: ev. Kirchensteuerpflicht
F17: rk. und ev. Kirchensteuerpflicht
F18: keine Kirchensteuerpflicht
F19: Altersentlastungsbetrag berücksichtigen
F2o: keine Arbeitslosenversicherung
F22: benutzt
F27: gesetzt
F29: "

Datenspeicher:

Roo: Bruttolohn
Ro1: Ecklohn
Ro2: Steuerpflichtiger Betrag
Ro3: Sozialversicherungspflichtiger Betrag
Ro4: VWL

Ro6: benutzt
Ro7: "
Ro8: "
Ro9: Anzahl Kinder
R1o: Steuerklasse, benutzt

Status:

SIZE o11, FIX 2, SF 27, CF 28, SF 29, Register: 179 = 2 Memory Module,

Das Programm wurde anhand der "Lohnsteuertabelle 198o", "Sozialversicherungs-Tabellen" vom Fachverlag Schäffer GmbH & Co, Stuttgart, und dem ESTG i.d.F.v. 21.o6.1979 und i.d.F.v. 25.o6.198o erstellt. Der Verfasser übernimmt keine Garantie für die Richtigkeit des Programms und lehnt jede Haftung, die bei Benutzung des Programms entsteht, ab. Das Programm darf außerhalb des Bereichs des HP-Anwender-Club Oliver Rietschel nur mit Zustimmung des Verfassers verbreitet werden.

Raimund Berg, Hagener Straße 2oo, 591o Kreuztal-Eichen

```
351 RDN
352 FS?C 18
353 CLX
354 FS? 17
355 XEQ 17
356 XEQ 22
357 "E KST="
358 ARCL X
359 FS? 16
360 PROMPT
361 "K KST="
362 ARCL X
363 FS? 15
364 PROMPT
365 ENTER↑
366 FS? 17
367 +
368 ST+ 01
369 RCL 03
370 XEQ 14
371 2,25
372 -
373 STO 03
374 440
375 X>Y?
376 SF 07
377 10
378 *
379 X<=Y?
380 STO 03
381 4
382 /
383 3
384 *
385 FS? 00
386 XEQ 23
387 X>Y?
388 X<>Y
389 5,5
390 FS? 07
391 CLX
392 FS? 08
393 CLX
394 %
395 RND
396 ST+ 01
397 "G="
398 ARCL X
399 PROMPT
400 RCL 03
401 9,25
402 FS? 07
403 CLX
404 %
405 RND
406 ST+ 01
407 "K="
408 FS? 00
409 "L="
410 ARCL X
411 PROMPT
412 CLX
413 1,5
414 FS?C 07
415 CLX
416 FS?C 20
417 CLX
418 %
419 RND
420 ST+ 01
421 "M="
422 ARCL X
423 PROMPT
424 RCL 04
425 ST+ 01
426 "VWL="
427 ARCL X
428 PROMPT
429 "ABZUEGE
=?"
430 ASTO 10
431 PROMPT
432 FS?C 22
433 ST+ 01
434 CLA
435 ARCL 10
436 "⊢E="
437 ARCL 01
438 PROMPT
439 RCL 00
440 RCL 01
441 -
442 "NETTO="
443 ARCL X
444 PROMPT
445 181,5
446 X<>Y
447 "ERS.K="
448 ARCL Y
449 FS? 08
450 PROMPT
451 FS?C 08
452 +
453 RCL 09
454 2
455 X<Y?
456 SF 07
457 RDN
458 RDN
459 RCL 04
460 FC? 07
461 ,3
462 FS?C 07
463 ,4
464 *
465 "SPARZUL
.="
466 ARCL X
467 PROMPT
468 +
469 "ERSTATT
UNG ?"
470 PROMPT
471 FS?C 22
472 +
473 "AUSZAHL
EN="
474 ARCL X
475 PROMPT
476♦LBL 16
477 4200
478 FS? 07
479 XEQ 17
480 FC? 03
481 XEQ 17
482 600
483 FS?C 07
484 XEQ 17
485 FS? 04
486 XEQ 17
487 RCL 09
488 *
489 +
490 RCL 08
491 X>Y?
492 X<>Y
493 RTN
494♦LBL 17
495 2
496 /
497 RTN
498♦LBL 19
499 2
500 /
501 XEQ 20
502 2
503 *
504 RTN
505♦LBL 18
506 54
507 /
508 INT
509 54
510 *
511 RTN
512♦LBL 22
513 1 E2
514 *
515 INT
516 1 E2
517 /
518 RTN
519♦LBL 23
520 X<=Y?
521 SF 08
522 RTN
523♦LBL 20
```

```
524 FC? 09
525 XEQ 18
526 STO 05
527 1 E4
528 /
529 6
530 X<>Y
531 X>Y?
532 SF 07
533 1,8
534 -
535 STO 06
536 4,2
537 -
538 FS? 07
539 STO 06
540 FS?C 07
541 GTO 24
542 18 E3
543 RCL 05
544 X<=Y?
545 GTO 26
546 RCL 06
547 3,05
548 *
549 73,76
550 -
551 RCL 06
552 *
553 695
554 +
555 RCL 06
556 *
557 2200
558 +
559 RCL 06
560 *
561 3034
562 +
563 INT
564 RTN
565♦LBL 24
566 13 E4
567 RCL 05
568 X>Y?
569 GTO 25
570 RCL 06
571 ,09
572 *
573 5,45
574 -
575 RCL 06
576 *
577 88,13
578 +
579 RCL 06
580 *
581 5040
582 +
583 RCL 06
584 *
585 20018
586 +
587 INT
588 RTN
589♦LBL 25
590 RCL 05
591 ,56
592 *
593 14837
594 -
595 INT
596 RTN
597♦LBL 26
598 4213
599 X<>Y
600 X>Y?
601 GTO 27
602 0
603 RTN
604♦LBL 27
605 ,22
606 *
607 926
608 -
609 INT
610 END
```

```
 01♦LBL "LST
"
 02 0
 03 STO d
 04 CLRG
 05 SF 29
 06 SF 27
 07 FIX 2
 08 "STPFL="
 09 ASTO 06
 10 "SOZV.="
 11 ASTO 07
 12 6
 13 "ST.KL?"
 14 PROMPT
 15 INT
 16 X<=0?
 17 LOG
 18 X>Y?
 19 ASIN
 20 SF IND X
 21 STO 10
 22 CF 22
 23 "KINDER?
"
 24 PROMPT
 25 FC?C 22
 26 0
 27 INT
 28 X<0?
 29 LOG
 30 STO 09
 31 SF 18
 32 "RELIGIO
N?"
 33 AON
 34 PROMPT
 35 AOFF
 36 ASTO 00
 37 "      "
 38 ARCL 00
 39 ASTO 00
 40 ASHF
 41 ASTO Z
 42 "  "
 43 ARCL 00
 44 ASHF
 45 ASTO X
 46 "K"
 47 ASTO Y
 48 X=Y?
 49 SF 15
 50 X=Y?
 51 CF 18
 52 "E"
 53 ASTO Y
 54 X=Y?
 55 SF 16
 56 X=Y?
 57 CF 18
 58 FS? 18
 59 GTO 07
 60 "K"
 61 FS? 15
 62 "E"
 63 ASTO Y
 64 RDN
 65 X≠Y?
 66 GTO 07
 67 SF 15
 68 SF 16
 69 SF 17
 70♦LBL 07
 71 CLX
 72 "ALTER?"
 73 PROMPT
 74 INT
 75 63
 76 X<Y?
 77 SF 19
 78 X<=Y?
 79 SF 20
 80 CF 22
 81 "GEHALT?
"
 82 PROMPT
 83 FS? 22
 84 STO 00
 85 FS? 22
 86 SF 00
 87 FS?C 22
 88 GTO 08
 89 "ECKLOHN
?"
 90 PROMPT
 91 X<0?
 92 LOG
 93 STO 01
 94 CF 22
 95 "GSTD.=?
"
 96 ASTO 08
 97 PROMPT
 98 FC?C 22
 99 0
100 X<0?
101 LOG
102 *
103 STO 00
104 CLA
105 ARCL 08
106 ARCL X
107 PROMPT
108♦LBL 09
109 "Z-STD=?
"
110 ASTO 08
111 PROMPT
112 FC?C 22
113 GTO 08
114 RCL 01
115 "% ?"
116 PROMPT
117 FC?C 22
118 GTO 08
119 %
120 RND
121 "%="
122 ARCL X
123 PROMPT
124 RCL Z
125 *
126 RND
127 CLA
128 ARCL 08
129 ARCL X
130 PROMPT
131 XEQ 13
132 GTO 09
133♦LBL 08
134 "VWL ANT
RAG?"
135 PROMPT
136 FC?C 22
137 GTO 10
138 X<0?
139 LOG
140 ST+ 00
141 +
142 LASTX
143 X<>Y
144 X<Y?
145 ASIN
146 52
147 X<>Y
148 X>Y?
149 ASIN
150 STO 04
151♦LBL 10
152 "GRATI.?
"
153 PROMPT
154 FS?C 22
155 XEQ 11
156♦LBL 12
157 "SONSTIG
ES?"
158 PROMPT
159 FS? 22
160 XEQ 13
161 FS?C 22
162 GTO 12
163 "FREIBET
RAG?"
164 PROMPT
165 FS?C 22
166 ST- 02
167 RCL 00
168 ST+ 02
169 ST+ 03
170 ,4
171 *
172 250
173 X>Y?
174 X<>Y
175 FS?C 19
```

```
176 ST- 02
177 CLA
178 ARCL 06
179 ARCL 02
180 PROMPT
181 CLA
182 ARCL 07
183 ARCL 03
184 PROMPT
185 "BRUTTO=
"
186 ARCL 00
187 PROMPT
188 RCL 02
189 XEQ 14
190 12
191 *
192 STO 07
193 9
194 %
195 4752
196 X>Y?
197 X<>Y
198 STO 08
199 GTO IND
10
200◆LBL 11
201 ST+ 00
202 600
203 ST- 02
204 6
205 /
206 X>Y?
207 X<>Y
208 ST- 03
209 RTN
210◆LBL 13
211 RDN
212 RDN
213 AON
214 CLA
215 ARCL 06
216 "⊢?"
217 PROMPT
218 ASTO X
219 "J"
220 ASTO Y
221 X=Y?
222 SF 07
223 CLA
224 ARCL 07
225 "⊢?"
226 PROMPT
227 ASTO X
228 X=Y?
229 SF 08
230 RDN
231 RDN
232 ST+ 00
233 FC?C 07
234 ST- 02
235 FC?C 08
236 ST- 03
237 AOFF
238 RTN
239◆LBL 14
240 3
241 +
242 4,5
243 /
244 INT
245 4,5
246 *
247 1,49
248 +
249 RTN
250◆LBL 01
251 0
252 STO 09
253◆LBL 04
254 1314
255 GTO 15
256◆LBL 02
257 RCL 09
258 X≠0?
259 SF 07
260 2136
261 ENTER↑
262 2178
263 FS?C 07
264 +
265 GTO 15
266◆LBL 03
267 1584
268◆LBL 15
269 STO 10
270 XEQ 16
271 SF 07
272 XEQ 16
273 RCL Z
274 +
275 XEQ 18
276 RCL 10
277 +
278 CHS
279 RCL 07
280 +
281 FS? 03
282 XEQ 19
283 FC? 03
284 XEQ 20
285 GTO 21
286◆LBL 05
287◆LBL 06
288 SF 09
289 0
290 STO 09
291 RCL 07
292 FS? 05
293 1044
294 FS? 06
295 18
296 -
297 XEQ 18
298 STO 07
299 STO 10
300 ,22
301 ST* 10
302 RDN
303 1,5
304 *
305 XEQ 19
306 X<> 07
307 2,5
308 *
309 XEQ 19
310 RCL 07
311 -
312 RCL 10
313 INT
314 X<=Y?
315 X<>Y
316◆LBL 21
317 1,2
318 /
319 INT
320 10
321 /
322 "LST="
323 ARCL X
324 PROMPT
325 STO 01
326 RCL 09
327 X=0?
328 SF 07
329 1
330 -
331 150
332 *
333 20
334 -
335 50
336 X>Y?
337 X<>Y
338 RDN
339 FS? 04
340 XEQ 17
341 FS?C 07
342 CLX
343 -
344 8
345 %
346 XEQ 22
347 X<=0?
348 SF 10
349 ,6
350 X<=Y?
```

W[illegible]ang Ehrhardt
Im gr[illegible] Feld 11
3400 Göttingen

Programmpaket INTERPOLATION

Inhalt:

1. Allgemeines

Es seien paarweise verschiedene Punkte x_o, $x_1, \ldots, x_n$ mit dazugehörigen Funktionswerten y_o, $y_1, \ldots, y_n$ gegeben. Gesucht wird ein Polynom vom Grade n (oder eine rationale Funktion) $f(x)$ mit $f(x_k) = y_k$ für k=0,..,n.
Das Programmpaket enthält verschiedene Unterprogramme zur Berechnung des Polynoms in unterschiedlichen Formen, sowie zur Ermittlung der Funktionswerte und Ableitungswerte für beliebige x-Werte.
Es werden im folgenden keine mathematischen Beweise angegeben, sondern nur die zum Verständnis unbedingt notwendigen Definitionen und Ergebnisse aufgeführt. Für weitergehende Informationen wird auf das Buch

[1] Meinardus,G.; G.Merz: Praktische Mathematik I
Mannheim, Wien, Zürich 1979 (Bibliographisches Institut)

verwiesen.

2. Polynominterpolation

Für das in Abschnitt 1 formulierte Interpolationsproblem gibt es ein eindeutig bestimmtes Polynom vom Grade n

(1) $p(x) = a_o + a_1 x + a_2 x^2 + \ldots + a_n x^n$.

Zur Berechnung dieses Polynoms werden sogenannte dividierte Differenzen Δ_i^k verwendet, die rekursiv wie folgt definiert sind:

$$\Delta_i^o = y_i \qquad (i=o,..,n)$$

(2) $$\Delta_i^k = \frac{\Delta_i^{k-1} - \Delta_{i+1}^{k-1}}{x_i - x_{i+k}}$$

Mit $b_k = \Delta_k^k$, k=o,..,n gilt dann für das Interpolationspolynom:

(3) $q(x) = b_0 + b_1(x-x_0) + .. + b_n(x-x_0)(x-x_1)..(x-x_{n-1})$.

Mit einem modifizierten Hornerschema (vgl. [1]) erhält man aus $q(x)$ eine weitere Darstellung des Interpolationspolynoms:

(4) $r_\xi(x) = c_0 + c_1(x-\xi) + .. + c_n(x-\xi)^n$.

Dabei ist ξ beliebig. Für $\xi = 0$ ergibt sich $p(x)$. Man beachte, daß $c_0 = r_\xi(\xi)$ und $c_1 = r_\xi'(\xi)$. Mit dem normalen Hornerschema wird $r_\xi(x)$ aus $p(x)$ berechnet. Die Unterprogramme zur Ermittlung von $p(\xi)$, $p'(\xi)$ (bzw. $q(\xi)$, $q'(\xi)$) verwenden verkürzte Formen des vollständigen normalen (bzw. modifizierten) Hornerschemas.

3. Rationale Interpolation

Für die rationale Interpolation wird der sogenannte Thielesche Kettenbruch verwendet, d.h. die rationale Interpolationsfunktion $R(x)$ sieht wie folgt aus:

$$R(x) = c_0 + \cfrac{x-x_0}{c_1 + \cfrac{x-x_1}{c_2 + \ddots \cfrac{x-x_{n-1}}{c_n}}} \qquad (5)$$

Zur Berechnung der c_k werden die sog. inversen Differenzen ϱ_i^k benutzt, die rekursiv wie folgt definiert sind:

$$\varrho_i^0 = y_i \;, \quad i=0,..,n$$

$$\varrho_i^{k+1} = \frac{x_i - x_k}{\varrho_i^k - \varrho_k^k} \qquad (6)$$

Falls die inversen Differenzen ϱ_k^k existieren und ungleich Null sind, gilt $c_k = \varrho_k^k$. Falls das nicht gilt, ist die rationale Interpolation in dieser Form nicht möglich.
Zur Berechnung von $R(\xi)$ wird der Kettenbruch (5) von "unten nach oben" ausgewertet.

4. Programmbeschreibungen

In den verschiedenen Unterprogrammen werden die HP 41 C - Register R00 bis R07 benutzt. Die Koeffizienten $\{a_k\}$, $\{b_k\}$ ... können ab Register R08 abgelegt werden. Die einzelnen Werte $\{..\}$ müssen fortlaufend gespeichert werden, z.B. $x_0 \to R08$, $x_1 \to R09$, $x_2 \to R10$..
Für die Rechnungen werden weiterhin die Anfangsadressen der Koef-

fizientenvektoren benötigt, so gibt z.B. $A(x_0)$ die Adresse von x_0 an, im obigen Beispiel also $A(x_0) = 8$. Was in welchem Register gespeichert ist, zeigt die folgende Tabelle. Ein ж zeigt dabei an, daß das entsprechende Register neben den bezeichneten ebenfalls benutzt wird. Die Bedeutung von i,j,k,l ist aus den Beschreibungen der benutzten Algorithmen ersichtlich.

UPrgrm	R00	R01	R02	R03	R04	R05	R06	R07
PM		n	$A(v_0)$	$A(w_0)$		i	j	
DD	ж	n	$A(y_0)$	$A(b_0)$	$A(x_0)$	i	ж	k
MH	ξ	n	$A(b_0)$	$A(c_0)$	$A(x_0)$	i	j	k
BM	ξ	n		$A(b_0)$	$A(x_0)$	i	j	
NH	ξ	n	$A(a_0)$	$A(c_0)$		ж	j	k
BN	ξ	n	$A(a_0)$			i		
ID	l	n	$A(y_0)$	$A(c_0)$	$A(x_0)$	i	j	k
BK	ξ	n		$A(c_0)$	$A(x_0)$	i	j	

4.1 Unterprogramm PM

Verschiebt $\{v_k\}$ nach $\{w_k\}$, benötigt $A(v_0)$, $A(w_0)$, n.
Wenn $A(v_0) = A(w_0)$ ist, wird nichts verschoben.

4.2 Unterprogramm DD

Berechnet aus $\{x_k\}$, $\{y_k\}$ die dividierten Differenzen $b_k = \Delta_k^k$.
Benötigt $A(x_0)$, $A(y_0)$, $A(b_0)$, n.

4.3 Unterprogramm MH

Modifiziertes Hornerschema. Berechnet aus $\{x_k\}$, $\{b_k\}$ in (3) die Werte $\{c_k\}$ in (4). Benötigt $A(x_0)$, $A(b_0)$, $A(c_0)$, n, ξ.

4.4 Unterprogramm BM

Berechnet $q(\xi)$ und $q'(\xi)$ in (3)
Benötigt $A(x_0)$, $A(b_0)$, n, ξ.

4.5 Unterprogramm NH

Normales Hornerschema. Berechnet $\{c_k\}$ in (4) aus $\{a_k\}$ in (1).
Benötigt $A(a_o)$, $A(b_o)$, n, ξ.

4.6 Unterprogramm BN

Berechnet $p(\xi)$ und $p'(\xi)$ aus (1).
Benötigt $A(a_o)$, n, ξ.

4.7 Unterprogramm ID

Berechnet aus $\{x_k\}$, $\{y_k\}$ die inversen Differenzen $\{c_k\}$ in (5).
Benötigt $A(x_o)$, $A(y_o)$, $A(c_o)$, n.

4.8 Unterprogramm BK

Berechnet den Thieleschen Kettenbruch $R(\xi)$ in (5).
Benötigt $A(x_o)$, $A(c_o)$, n, ξ.

Es ist im allgemeinen möglich, daß die zu berechnenden Koeffizienten die alten Werte überschreiben, z.B. ist im Unterprogramm DD $A(b_o) = A(y_o)$ möglich. Dies spart zwar Speicherplatz, nur sind dann die alten Werte natürlich nicht mehr ansprechbar.

5. Algorithmen

Im folgenden werden für die im Abschnitt 4 beschriebenen Unterprogramme die benutzten Algorithmen angegeben. Die verwendete Notation dürfte allgemein verständlich sein, so bedeutet z.B. die Schreibweise " for k = i(j)n ", daß entsprechend dem BASIC-Befehl " FOR K = I TO N STEP J " verfahren wird. Die Algorithmen orientieren sich stark an den HP 41 C-Möglichkeiten; für eine Übertragung in höhere Programmiersprachen kann man sie noch vereinfachen. Die Schreibweise " $PM(v_o, w_o, n)$ " bedeutet, daß die n+1 Werte $\{v_k\}$ mit dem Unterprogramm PM nach $\{w_k\}$ verschoben werden.
Die Unterprogramme BM und BN liefern die Werte $q(\xi)$, $q'(\xi)$ (bzw. $p(\xi)$, $p'(\xi)$) in den Stackregistern X und Y. $R(\xi)$ wird von BK in X geliefert.

BM und BN liefern neben den Funktionswerten des Interpolationspolynoms auch die Werte der ersten Ableitung. Wem die angegebenen Unterprogramme zu langsam oder zu speicherplatzaufwendig sind, kann leicht an Hand der Algorithmenlistings diese so modifizieren, daß sie nur noch die Funktionswerte berechnen.

PM

```
if  A(v_o) = A(w_o)  return
j=0
for i=0(1)n
  w_i=v_j
  j=j+1
next i
```

MH

```
PM(b_o,c_o,n)
for k=n(-1)1
  i=n; x=c_i; i=i-1
  for j=k-1(-1)0
    x=(ξ-x_j)*x + c_i
    c_i=x; i=i-1
  next j
next k
```

DD

```
PM(y_o,b_o,n)
for k=1(1)n
  y=b_k-1
  for i=k(1)n
    x=(y-b_i)/(x_i-k-x_i)
    y=b_i; b_i=x
  next i
next k
```

BK

```
x=c_n; j=n-1
for i=n-1(-1)0
  x=(ξ-x_i)/x + c_j
  j=j-1
next i
```

BN

```
y=0
x=a_n
for i=n-1(-1)0
  y=y*ξ + x
  x=x*ξ + a_i
next i
```

BM

```
j=n; y=0; x=b_n; j=j-1
for i=n-1(-1)0
  y=(ξ-x_i)*y + x
  x=(ξ-x_i)*x + b_j
  j=j-1
next i
```

ID

```
PM(y_o,c_o,n)
for i=0(1)n-1
  l=i; k=l+1
  for j=i+1(1)n
    c_k=(x_i-x_j)/(c_l-c_k)
    k=k+1
  next j
  l=l+1
next i
```

NH

```
PM(a_o,c_o,n)
for j=0(1)n
  x=0
  for k=n(-1)j
    x=c_k+ξ*x; c_k=x
  next k
next j
```

6. HP 41 C - Programme

```
LBL 'PM
RCL 02  RCL 03  X=Y?  RTN
RCL 01  +  1E3  /  RCL 03
+ STO 05 RCL 02  .9  + STO 06
LBL 01
  RCL IND 06  STO IND 05
  ISG 06  ISG 05  GTO 01
END
```

```
LBL 'DD
XEQ 'PM  RCL IND 03  STO 00
RCL 03  RCL 01  +  1E3  /
RCL 03  +  1  +  STO 07
LBL 01
  RCL 04  .9  +  STO 06
  RCL 07  STO 05  +
  RCL 03  -  X<>00
    LBL 02
    RCL IND 05  -  RCL IND 06
    RCL IND 00  -  /
    X<>IND 05  ISG 00
    ISG 06  ISG 05  GTO 02
  RCL IND 07  STO 00
  ISG 07  GTO 01
END
```

```
LBL 'MH
XEQ 'PM  RCL 01  STO 07
RCL 04  1  -  .1  %  +
STO 06
LBL 02
  RCL 03  RCL 01  +  STO 05
  RCL 07  ST+ 06
  RCL IND 05  DSE 05
  LBL 03
    RCL 00  RCL IND 06  -  *
    RCL IND 05  +  STO IND 05
    DSE 05  DSE 06  GTO 03
  DSE 07  GTO 02
END
```

```
LBL 'BM
RCL 03  RCL 01  +  STO 06
RCL 04  1  -  .1  %  + RCL 01
+  STO 05  0  RCL IND 06  DSE 06
LBL 01
  RCL 00  RCL IND 05  -  ST* Z
  X<>Y  ST+ Z  *  RCL IND 06  +
  DSE 06  DSE 05  GTO 01
END
```

```
LBL 'NH
XEQ 'PM  RCL 03  RCL 01  +
1E3  /  RCL 03  +  STO 06
LBL 01
  RCL 03  RCL 01  +  STO 07
  RCL 06  STO 05  RCL 00
  ↑  ↑  ↑  CLX
  LBL 02
    *  RCL IND 07 +
    STO IND 07  DSE 07
    ISG 05  GTO 02
  ISG 06  GTO 01
END
```

```
LBL 'BN
RCL 03  1  -  1E3  /  RCL 03
RCL 01  +  +  STO 05
0  RCL IND 05  DSE 05
  LBL 01
  RCL 00  ST* Z  X<>Y  ST+ Z
  *  RCL IND 05 +  DSE 05  GTO 01
END
```

```
LBL 'ID
XEQ 'PM  RCL 04  XEQ 04  STO 05
RCL 03  XEQ 04  STO 00
LBL 01
  RCL 05  1.001  +  STO 06
  RCL 00  LASTX  +  STO 07
  LBL 02
    RCL IND 05  RCL IND 06  -
    RCL IND 00  RCL IND 07  -  /
    STO IND 07  ISG 07  LBL 03
    ISG 06  GTO 02
  ISG 00  LBL 03  ISG 05  GTO 01
RTN
LBL 04  STO Y  RCL 01  +  1  -
1E3  /  +
END
```

```
LBL 'BK
RCL 04  1  -  .1  %  +  RCL 01  +
STO 05  RCL 03  RCL 01  +  STO 06
  RCL IND 06  DSE 06
  LBL 01
    1/X  RCL 00  RCL IND 05  -
    *  RCL IND 06  +  DSE 06
    DSE 05  GTO 01
  END
```

Bemerkungen: ↑ steht für ENTER ↑
* steht für Multiplikation

7. Beispiele

Gegeben seien die folgenden x- und y-Werte:

x	-2	-1	0	1	2
y	81	16	1	0	1

Man löse mit Polynominterpolation folgende Aufgaben:

a. Bestimme $q(x)$, d.h. $\{b_k\}$ in (3), sowie $q(.5)$ und $q'(.5)$
b. Bestimme $p(x)$, d.h. $\{c_k\}$ für $\xi=0$, $p(.5)$ und $p'(.5)$
c. Bestimme $p_1(x)$, d.h. $\{c_k\}$ für $\xi=1$

Lösungen: a. Es ist n=4, wähle $A(x_0)=10$, $A(y_0)=15$ und $A(b_0)=20$.

```
-2 STO 10   -1 STO 11   0 STO 12   1 STO 13   2 STO 14
81 STO 15   16 STO 16   1 STO 17   0 STO 18   1 STO 19
 4 STO 01   10 STO 04  15 STO 02  20 STO 03
```

Die $\{b_k\}$ erhält man dann mit XEQ 'DD
$b_0=81$, $b_1=-65$, $b_2=25$, $b_3=-6$, $b_4=1$
.5 STO 00 XEQ 'BM ergibt : $q(.5)=0.0625=1/16$, $q'(.5)=-.5=-1/2$

b. Die $\{c_k\}$ sollen die $\{b_k\}$ überschreiben, es ist $\xi=0$.
20 STO 02 0 STO 00 XEQ 'MH ergibt:
$c_0=1$, $c_1=-4$, $c_2=6$, $c_3=-4$, $c_4=1$, d.h. für $p(x)$ erhält man:
$p(x) = 1 - 4x + 6x^2 - 4x^3 + x^4$
.5 STO 00 XEQ 'BN ergibt wieder $p(.5)=o.o625$ und $p'(.5)=-o.5$

c. Die neuen $\{c_k\}$-Werte sollen wieder die alten überschreiben.
1 STO 00 XEQ 'NH ergibt $p_1(x) = (x-1)^4$.

Ein zweites Beispiel soll die rationale Interpolation behandeln. Dazu seien die folgenden x- und y-Werte gegeben:

x	-2	-1	0	1	2
y	.2	0	1	2	1.8

Es soll der Thielesche Kettenbruch, d.h. die $\{c_k\}$ in (5), berechnet werden, außerdem ist nach interpolierten y-Werten für die x-Werte -1.5, -0.5, 0.5, 1.5 gefragt.

Lösung: Es ist wieder n=4. Wähle $A(x_0)=8$, $A(y_0)=14$, $A(c_0)=20$.

```
-2 STO 08   -1 STO 09   0 STO 10   1 STO 11    2 STO 12
.2 STO 14    0 STO 15   1 STO 16   2 STO 17  1.8 STO 18
 4 STO 01    8 STO 04  14 STO 02  20 STO 03
```

Mit XEQ 'ID erhält man jetzt die Koeffizienten $\{c_k\}$

$c_0 = 0.2000 = 1/5$ $\qquad R(x) = \frac{1}{5} + \cfrac{x+2}{-5 + \cfrac{x+1}{\frac{2}{15} + \cfrac{x}{6 + \cfrac{x-1}{2/3}}}}$

$c_1 = -5.0000 = -5$

$c_2 = 1.3333 = 2/15$

$c_3 = 6.0000 = 6$

$c_4 = 0.6667 = 2/3$

Berechnet man den Kettenbruch von "unten nach oben", so ergibt sich nach einigen Umformungen die interpolierende rationale Funktion R(x)

$$R(x) = \frac{1 + 2x + x^2}{1 + x^2} .$$

Die geforderten interpolierten Werte werden wie folgt berechnet:

```
-1.5 STO 00  XEQ 'BK        R(-1.5) = 0.0769
-0.5 STO 00  XEQ 'BK        R(-0.5) = 0.2000
 0.5 STO 00  XEQ 'BK        R( 0.5) = 1.8000
 1.5 STO 00  XEQ 'BK        R( 1.5) = 1.9231
```

Zu den angegebenen Beispielen: Sie wurden so gewählt, daß die zu interpolierenden Funktionen von den entsprechenden Unterprogrammen reproduziert werden. Das ist didaktisch vielleicht nicht besonders geschickt, zeigt aber, daß die Unterprogramme richtig arbeiten.

Anmerkungen

1. Polynominterpolation ist immer möglich, falls die $\{x_k\}$ paarweise verschieden sind. Wenn n groß ist, kann es aber sein, daß (besonders nahe der Endpunkte) starke Oszillationen auftreten. Das kann durch Zeichnen der Funktion kontrolliert werden. Bei der rationalen Interpolation kann es vorkommen, daß Differenzen verschwinden und ID/BK mit 'DATA ERROR' (Division durch 0) abbrechen.

2. Selbstverständlich müssen nicht unbedingt alle Unterprogramme gleichzeitig im Speicher sein. Außerdem gibt es natürlich weitere Möglichkeiten Speicherplatz zu sparen: Benötigt man z.B. bei der rationalen Interpolation die y-Werte nicht weiter, so kann man $A(y_0)$ in R03 speichern und löscht den Schritt "XEQ 'PM" in ID. Das Unterprogramm PM wird dann nicht benötigt, und die $\{c_k\}$ überschreiben die $\{y_k\}$.

Niels Nöhren
Kielort 16
2000 Norderstedt
F: (040) 524 28 88

Norderstedt, 01.12.198

Herrn
Oliver Rietschel
Hewlett-Packard
Anwender-Club
Postfach 373

2420 Eutin

Liebe Clubfreunde!

Wir leben in einer Zeit, in der ständig steigende Benzinpreise und andere Faktoren das Autofahren immer kostspieliger machen.

Es kann nützlich sein, den Benzinverbrauch und die Kostenentwicklung seines Wagens zu beobachten, um daraus Rückschlüsse auf das Fahrverhalten ziehen zu kömnen.

Das Programm "Car Control" ist bei dieser Beobachtung ein guter Helfer.

Nach Eingabe aller notwendigen Fakten errechnet das Programm:

1. Fahrleistung per Monat, per Jahr und total.

2. Benzinverbrauch per 100 km zwischen dem letzten und dem jüngsten Tanken, per Monat, per Jahr und total.

3. Betriebskosten je 100 km per Monat, per Jahr und total.

4. Fixkosten je 100 km total und Gesamtkosten je 100 km total.

5. Rücklagen je Monat für Werkstattkosten und Bereifung.

6. Aktueller Restwert des Wagens nach einer Formel des ADAC.

Configuration:

Size: 034
Programmschritte: 383
Flags: Ø, 1, 2, 22, 25, 28, 29.
Register: 18 bis 33.

Ich benutze die angegebenen Register, damit die Daten nach Möglichkeit im Rechner gespeichert bleiben können. Die Register enthalten:

R 18: Summe der Betriebskosten des laufenden Monats.
R 19: Menge des getankten Benzins im laufenden Monat.
R 20: letzter Kilometerstand.
R 21: Kaufdatum des Wagens.
R 22: letztes Eingabedatum.
R 23: Kilometerstand beim letzten Tanken.
R24: Kilometerleistung im laufenden Monat.
R 25: Zeitspanne in Jahren zwischen Kaufdatum und letztem Eingabedatum.
R 26: Kilometerleistung seit Kaufdatum.
R 27: Menge des getankten Benzins seit Kauf.
R 28: Summe der Betriebskosten seit Kauf.
R 29: Summe der Fixkosten seit Kauf.
R 30: Summe der Betriebskosten des laufenden Jahres.
R 31: Kaufpreis des Wagens.
R 32: Benzinmenge des laufenden Jahres.
R 33: Kilometerleistung des laufenden Jahres.

Die Gesamtkosten ergeben sich aus den Betriebskosten, den Fixkosten und den buchmäßigen Kosten. Die buchmäßigen Kosten setzen sich zusammen aus Grundabschreibung, Wertverlust und Zinsverlust des Anschaffungswertes.

Laut dem ADAC kann man allgemein als Grundabschreibung 15% p.a. und als Wertverlust 25 Promille per 1000 km rechnen. Für den Zinsverlust habe ich eine mögliche Verzinsung von 6% p.a. des Anschaffungswertes angenommen.

Das Programm wird mit "XEQ CC" gestartet. Die Anzeige "DD.MM.YYYY ?" fragt, ob das zuletzt eingegebene Datum noch gilt. Wird ein neues Datum eingegeben (bitte unbedingt in der Form DD,MMYYYY), prüft der Rechner das Datum zunächst auf Legalität um dann die Zeitspanne in Jahren seit Kaufdatum zu berechnen und abzuspeichern.

Die nächste Frage gilt dem Kilometerstand. Zum Beispiel "KM 71005 ?".

Werden weder ein neues Datum noch ein neuer Kilometerstand eingegeben, geht der Rechner zu jeweils nächsten Programmpunkt über.

Bei Eingabe eines neuen Kilometerstandes wird die Differenz zum letzten ermittelt und zur Fahrleistungssumme des laufenden Monats, des laufenden Jahres und zur totalen Fahrleistungs-

...

summe addiert.

Mit "AUSGABE ?" fragt der Rechner, ob eine Datenausgabe erwünscht ist. Hier und bei der Anzeige "EINGABEN ?" steht der Rechner im ALPHA-Modus. Der Rechner erwartet jetzt eine bejahende oder verneinende Antwort. Verneint wird mit der Eingabe "N" und R/S. Bejaht wird mit "J", einer beliebigen Antwort oder nur mit R/S.

Nehmen wir an, es sind Eingaben zu machen. Jetzt erscheint "LTR ?". Hier wird die Tankmenge eingetippt.

Nach R/S wird der Durchschnittsverbrauch seit dem letzten Tanken errechnet. Anschließend wird die totale Gesamtmenge und der totale Durchschnitt angezeigt.

Mit "LAK ?" wird nach leistungsabhängigen Kosten oder Betriebskosten gefragt. Nach der Eingabe werden die totale Summe und Durchschnitt angezeigt.

"ZAK ?" steht für zeitabhängige Kosten oder Fixkosten. Auch hier der gleiche Ablauf.

Hier ist der Programmteil Eingabe beendet. Jetzt folgt der Ausgabeteil, falls erwünscht. Sollte keine Ausgabe erfolgen, schaltet der Rechner sich jetzt ab.

Bei der Ausgabe bedeuten:

"LTR/M="	Benzinmenge im laufenden Monat.
"X="	Durshschnitt auf 100 km.
"LTR/J="	Benzinmenge im laufenden Jahr.
"X="	Durchschnitt.
"LAK/M="	Betriebskosten im laufenden Monat.
"X="	Durchschnitt.
"LAK/J="	Betriebskosten im laufenden Jahr.
"X="	Durchschnitt.
"RW="	Restwert nach ADAC-Formel.
"GK="	Gesamtkosten total seit Kauf.
"X="	Durchschnitt.
"KM/X="	Kilometer im Durchschnitt per Jahr.
"KM/J="	Kilometer bisher im laufenden Jahr.
"KM/M="	" " " " Monat.
"WR/M="	Rücklage für Werkstattkosten per laufenden Monat.

Bei Ablauf eines Monats werden die Register für die Monatssummen auf Null gestellt. Gleiches gilt für die Register der Jahressummen bei Ablauf eines Jahres.

Bei Anschluss eines Kartenlesers kann man über "LBL G" die Daten auf Magnetkarte schreiben und über "LBL H" von Magnetkarte einlesen.

Da ich noch keinen Drucker besitze, habe ich das Programm natürlich noch nicht für Drucker ausgelegt.

Bei der erstmaligen Benutzung des Programms ist es natürlich wichtig, alle Grunddaten von Hand abzuspeichern.

Das Kaufdatum nach R 21. Den Kaufpreis nach R 31. Die bisherige Kilometerleistung nach R 20 und R 26. Den Kilometerstand vom letzten Tanken nach R 23. Die bisherige Tankmenge nach R 27. Die bisherigen Betriebskosten nach R 28. Die bisherigen Fixkosten nach R 29.

Beim Kauf eines neuen Autos sind ja nur 3 Grunddaten abzuspeichern. Das Kaufdatum, den Kaufpreis und der Kilometerstand bei Übernahme.

Ich hoffe, daß jemand dieses Programm gebrauchen kann und gute Programmverbesserungen würde ich gerne übernehmen.

Happy Programming

Euer

Niels

Programmlisting 1, Car Control, by Niels Nöhren.

```
001 LBL CC        049 LBL Ø3             095 ST+ 18
002 SF 28         050 " AUSGABE ?"       096 ST+ 28
003 SF 29         051 XEQ Ø1             097 RCL 28
004 RCL 22        052 SF Ø1              098 "LAK="
005 XEQ Y         053 X=Y?               099 ARCL X
006 FIX Ø         054 CF Ø1              100 PROMPT
007 " "           055 " EINGABEN ?"      101 RCL 26
008 ARCL Y        056 XEQ Ø1             102 XEQ Ø9
009 ARCL X        057 X=Y?
010 CF 28         058 GTO Ø6             103 LBL Ø5
011 CF 29                                104 CF 22
012 ARCL Z        059 LBL 1Ø             105 " ZAK ?"
013 ≡)- ?"        060 CF 22              106 PROMPT
014 CF 22         061 " LTR ?"           107 FC? 22
015 PROMPT        062 PROMPT             108 GTO Ø6
016 FC? 22        063 FC? 22             109 CF ØØ
017 GTO Ø2        064 GTO Ø4             110 ST+ 29
018 XEQ U         065 ST+ 32             111 RCL 29
019 STO 25        066 ST+ 19             112 "ZAK="
020 X()Y          067 ST+ 27             113 ARCL X
021 STO 22        068 X=Ø?               114 PROMPT
022 RCL 21        069 GTO Ø7             115 RCL 26
023 XEQ V         070 1 E2               116 XEQ Ø9
024 ST- 25        071 x
025 265,25        072 RCL 2Ø             117 LBL Ø6
026 ST/ 25        073 ENTER              118 FS? Ø1
                  074 X() 23             119 GTO ØØ
027 LBL Ø2        075 -                  120 FS? ØØ
028 RCL 22        076 /                  121 GTO 12
029 1             077 " X="              122 GTO G
030 +             078 ARCL X
031 SF ØØ         079 PROMPT             123 LBL ØØ
032 SF Ø3                                124 RCL 19
033 XEQ U         080 LBL Ø7             125 "LTR/M="
034 " KM"         081 RCL 27             126 ARCL X
035 ARCL 2Ø       082 "LTR="             127 PROMPT
036 ")- ?"        083 ARCL X             128 RCL 24
037 FIX 2         084 PROMPT             129 XEQ Ø9
038 SF 29         085 RCL 26             130 RCL 32
039 CF 22         086 XEQ Ø9             131 "LTR/J="
040 PROMPT                               132 ARCL X
041 FC? 22        087 LBL Ø4             133 PROMPT
042 GTO Ø3        088 CF 22              134 RCL 33
043 ENTER         089 " LAK ?"           135 XEQ Ø9
044 X() 2Ø        090 PROMPT             136 RCL 18
045 -             091 FC? 22             137 "LAK/M="
046 ST+ 33        092 GTO Ø5             138 ARCL X
047 ST+ 24        093 CF ØØ              139 PROMPT
048 ST+ 26        094 ST+ 3Ø             140 RCL 24
```

Man möge mir bitte verzeihen, wenn ich aus Platzgründen die dritte Seite auch noch auf dieses Blatt quetsche....Oliver

Programmlisting 3, Car Control

```
284 X≠Y?              332 x
285 CF Ø2             333 X()Y
286 X(Y?              334 RDN
287 GTO W             335 X()Y
288 CLX               336 R/
289 1                 337 RTN
290 +
291 RCL Z             338 LBL V
292 1 E4              339 XEQ Y
293 /                 340 1
294 +                 341 +
295 1 E2              342 1/X
296 /                 343 ,7
297 +                 344 ST+ X
298 XEQ V             345 X() L
299 STO Ø5            346 X()Y
300 RCL Ø4            347 CHS
301 XEQ V             348 XEQ O
302 ENTER             349 3Ø,6
303 X() Ø5            350 x
304 X(=Y?             351 INT
305 FC? ØØ            352 +
306 FS?C ØØ           353 X()Y
307 RTN               354 1 E2
308 X(=Y?             355 /
309 GTO W             356 INT
310 X()Y              357 7
311 +                 358 ST- Y
312 CLST              359 CLX
313 RCL Ø4            360 ,75
314 LASTX             361 STx Y
315 RTN               362 X() L
                      363 X()Y
316 LBL W             364 INT
317 CLST              365 ST- Z
318 RCL Ø4            366 CLX
319 FIX 6             367 36525
320 "DATUMSFEHLER"    368 x
321 PROMPT            369 INT
                      370 +
322 LBL Y             371 RTN
323 INT
324 ST- L             372 LBL O
325 LASTX             373 INT
326 1 E2              374 ST+ T
327 x                 375 X()Y
328 INT               376 INT
329 ST- L             377 CLX
330 LASTX             378 12
331 1 E4
                      379 STx Y
                      380 X() L
                      381 X()Y
                      382 -
                      383 END
```

Programmlis[illegible]g 2, Car Control, by Niels Nöhren

```
141 XEQ Ø9
142 RCL 3Ø
143 "LAK/J="
144 ARCL X
145 PROMPT
146 RCL 33
147 XEQ Ø9
148 1
149 RCL 26
150 1 E5
151 /
152 25
153 x
154 RCL 25
155 15
156 x
157 +
158 %
159 -
160 RCL 31
161 x
162 LASTX
163 X()Y
164 RND
165 X(Ø?
166 CLX
167 "RW="
168 ARCL X
169 PROMPT
170 -
171 1,Ø6
172 RCL 25
173 Y↑X
174 RCL 31
175 x
176 RND
177 RCL 31
178 -
179 +
180 RCL 28
181 +
182 RCL 29
183 +
184 "GK="
185 ARCL X
186 PROMPT
187 RCL 26
188 XEQ Ø9
189 CF 29
190 FIX Ø
191 RCL 26
192 RCL 25
193 /
194 "KM/X="
195 ARCL X
196 PROMPT
197 "KM/J="
198 ARCL 33
199 PROMPT
200 RCL 24
201 "KM/M="
202 ARCL X
203 PROMPT
204 FIX 2
205 SF 29
206 RCL 31
207 1 E3
208 /
209 SQRT
210 3,32
211 +
212 1 E2
213 /
214 x
215 "WR/M="
216 ARCL X
217 PROMPT

218 LBL 12
219 FC? ØØ
220 CF Ø2
221 FC?C ØØ
222 GTO G
223 CLX
224 STO 19
225 STO 18
226 STO 24
227 FC?C Ø2
228 GTO G
229 STO 33
230 STO 32
231 STO 3Ø

232 LBL G
233 SF 25
234 18,Ø33
235 WDTAX

236 LBL K
237 CF ØØ
238 CF Ø1
239 CF Ø2
240 CF Ø3
241 CF Ø4
242 CF 27
243 CF 28
244 SF 29
245 FIX 2
246 CLA
247 CLST
248 -
249 OFF
250 GTO K

251 LBL H
252 18,Ø33
253 RDTAX
254 GTO CC

255 LBL Ø9
256 SF 25
257 /
258 1 E2
259 x
260 " X="
261 ARCL X
262 PROMPT
263 RTN

264 LBL Ø1
265 AON
266 PROMPT
267 AOFF
268 ASTO Y
269 "N"
270 ASTO X
271 RTN

272 LBL U
273 STO Ø4
274 XEQ Y
275 X()Y
276 X=Ø?
277 GTO W
278 CLX
279 1
280 X()Y
281 X=Ø?
282 GTO W
283 12
```

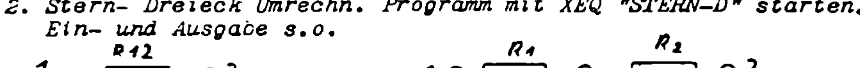
C Programm: Stern - Dreieck Umrechnung

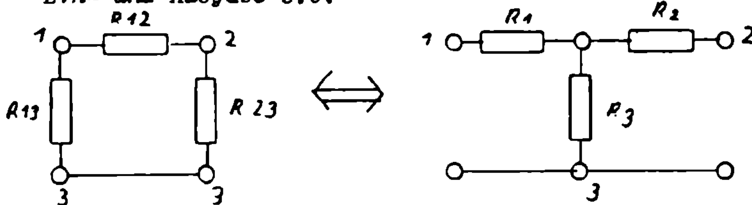
eibung: Dieses Programm berechnet die Ohmwerte von Widerständen, die von der Dreieckschaltung in die Sternschaltung umgeschaltet werden, und umgekehrt.

er: 25

er: STO o1 bis STO o7

rung: 1. Dreieck- Stern Umrechn. Programm mit XEQ "DREI-ST" starten, dann Dreieck- Widerstände nach Alpha- Abfrage mit R/S eingeben und Stern-Widerst. mit R/S in die Anzeige rufen.
2. Stern- Dreieck Umrechn. Programm mit XEQ "STERN-D" starten. Ein- und Ausgabe s.o.





g:

```
L "DREI-ST"
 12?"     Eingabe der
OMPT      Dreieck wider-
0 o1      stände
23?"
OMPT
0 o2
 31?"
OMPT
0 o3
L "A"     Berechnung
L o1      und Aus-
L o2      gabe der
          Stern-Widerst.
L o3

0 o7
L o1
L o3

L o7

  o4
1="
CL o4
OMPT
L o1
L o2

o3o RCL o7
o31 /
o32 STO o5
o33 "R2="
o34 ARCL o5
o35 PROMPT
o36 RCL o2
o37 RCL o3
o38 *
o39 RCL o7
o4o /
o41 STO o6
o42 "R3="
o43 ARCL o6
o44 PROMPT
o45 RTN
o46 LBL "STERN-D"
o47 "R1?"      Eingabe der
o48 PROMPT     Stern-Widerst.
o49 STO o4
o5o "R2?"
o51 PROMPT
o52 STO o5
o53 "R3?"
o54 PROMPT
o55 STO o6
o56 LBL "B"  Berechnung und
o57 RCL o4   Ausgabe der Drei-
o58 RCL o5   eck-Widerstände

o59 *
o6o RCL o5
o61 RCL o6
o62 *
o63 RCL o6
o64 RCL o4
o65 *
o66 +
o67 +
o68 STO o7
o69 RCL o6
o7o /
o71 STO o1
o72 "R12="
o73 ARCL o1
o74 PROMPT
o75 RCL o7
o76 RCL o4
o77 /
o78 STO o2
o79 "R23="
o8o ARCL o2
o81 PROMPT
o82 RCL o7
o83 RCL o5
o84 /
o85 STO o3
o86 "R31="
o87 ARCL o3
o88 PROMPT
o89 END
```

Jörg Meyer

2833 Harpstedt, 17.1.81
Lange Str. 54

HP 41-C Programm: Addition von 2 Sinusschwingungen gleicher Frequenz

Beschreibung: Dieses Programm addiert 2 Sinusförmige Schwingungen gleicher Frequenz zu ihrer resultierenden Schwingung.

Beispiel 1. $A1\ SIN(\omega t + \varphi_1) + A2\ SIN(\omega T + \varphi_2) = A\ SIN(\omega T + \varphi)$

2. $A1\ COS(\omega T + \varphi_1) + A2\ COS(\omega T + \varphi_2) = A\ COS(\omega T + \varphi)$

3. $A1\ COS(\omega T + \varphi_1) + A2\ SIN(\omega T + \varphi_2) = A\ SIN(\omega T + \varphi)$

Bei diesem Beispiel muß man zunächst die COS Schwingung in eine Sinusschwingung umwandeln. Das macht man nach folgender Formel:

$$COS(\omega T + \varphi_1) = SIN(\omega T + \varphi_1 + 90^\circ)$$

$$SIN(\omega T + \varphi_1) = COS(\omega T + \varphi_1 - 90^\circ)$$

Es ergibt sich also folgende Gleichung:

3. $A1\ SIN(\omega T + \varphi_1 - 90^\circ) + A2\ SIN(\omega T + \varphi_2) = A\ SIN(\omega T + \varphi)$

Register: 14

Speicher: STO oo bis STO o5

Ausführung: Programm mit XEQ "ADDITH" starten, dann Amplitude 1 und 2 und Winkel 1 und 2 mit R/S nach Alpha Abfrage eingeben, Ausgabe von der resultierenden Amplitude und dem resultierenden Nullphasenwinkel ebenfalls mit R/S.

Listing:

```
oo1 LBl "ADDITH"
oo2 LBL "A"      Eingabeprogramm
oo3 "A1=?"
oo4 PROMPT
oo5 STO oo
oo6 "A2=?"
oo7 PROMPT
oo8 STO o1
oo9 "∢1=?"
o1o PROMPT
o11 STO o2
o12 "∢2=?"
o13 PROMPT
o14 STO o3
-----------------
o15 LBL "B"      Berechnung der resultierenden Amplitude
o16 RCL oo
o17 X↑2
o18 RCL o1
o19 X↑2
o2o +
o21 RCL oo
o22 RCL o1
o23 *
o24 2
o25 *
o26 RCL o2
o27 RCL o3
o28 -
o29 COS
o3o *
o31 +
o32 SQRT
o33 STO o4
o34 "A="
o35 ARCL o4
o36 PROMPT
-----------------
o37 LBL "C"      Berechnung des resultierenden Nullphasenwinkels
o38RCL o2
o39 SIN
o4o RCL oo
o41 *
o42 RCL o3
o43 SIN
o44 RCL o1
o45 *
o46 +
o47 RCL o2
o48 COS
o49 RCL oo
o5o *
o51 RCL o3
o52 COS
o53 RCL o1
o54 *
o55 +
oo56 /
o57 ATAN
o58 STO o5
o59 "∢="
o6o ARCL o5
o61 PROMPT
o62 END
```

Flags und der Drucker

Funktion:	55 clear; 21 set	55 clear; 21 clear
ACA	X	/
ACCHR	X	/
ACCOL	X	/
ACSPEC	X	/
ACX	X	/
ADV	/	/
BLDSPEC	X	X
PRA	4	/
PRAXIS	1	/
PRBUF	/	/
PRFLAGS	2	/
PRKEYS	2	/
PRPLOTP	1	/
PRREG	3	/
PRREGX	3	/
PRΣ	3	/
PRSTK	3	/
PRX	4	/
REGPLOT	5	/
SKPCHR	X	/
SKPCOL	X	/
STKPLOT	5	/

Erprobt wurde diese Tabelle nur auf einem Drucker Serial No. 2o373S4oo54. Änderungen bei anderen Modellen sind also durchaus möglich.

Rolf Mach
Thorwaldsenstr. 3
6090 Rüsselsheim

Die in der Tabelle " Flags und der Drucker " verwendeten Symbole haben folgende Bedeutung:

X) Die Funktion arbeitet normal.

/) Die Funktion hat keine Wirkung.

1) Aufgrund falscher Formatierung ist der Ausdruck unbrauchbar.

2) Im Buffer stehende Zeichen werden ausgedruckt. Nach einiger "Bedenkzeit" passiert schließlich nichts.

3) Der Ausdruck ist in einem gestauchtem Format. Siehe Bsp.

4) Im Buffer stehende Zeichen werden ausgedruckt. Danach die Funktion ACA/ACX ausgeführt.

5) An den bisherigen Bufferinhalt werden nnn (Spaltenbreite des Plots; siehe Handbuch Seite 59 unten) leere Spalten angehängt. Bei Überlauf wird eine Zeile ausgedruckt.

Beispiel zu 3:

Im Buffer stehende Zeichen werden ausgedruckt, dann wird er mit den betreffenden Zahlen und Zeichen vollgeschrieben.

Normal, groß

```
R00=   3,1416
R01=   3,0000
R02=   28,500
0
R03=  "*REG3*
"
R04=  -523,22
24
R05=  -0,0023
R06=   4,0000
```

Gestaucht, groß

```
R 3,1416R 3,
0000R 28,500
0R*REG3*R-52
3,2224R-0,00
  23R 4,0000
```

Normal, klein

```
R00=  3,1416
R01=  3,0000
R02=  28,5000
R03= "*REG3*"
R04= -523,2224
R05= -0,0023
R06=  4,0000
```

Gestaucht, klein

```
R 3,1416R 3,0000R 28,500
0R*REG3*R-523,2224R-0,00
              23R 4,0000
```

Rüdiger Schultz
Villa Viktoria
CH-9402-MOERSCHWIL
Schweiz

23. februar I981

Hallo Clubmitglieder
S
Seit etwa zwei Monaten bin ich stolzer Besitzer eines HP 4IC und zwei RAM's. kartenles er und Barcodeleser sind bestellt.
Mein Beruf:Futtermüller.Jahrgang:I962.Hobbies:Astronomie,Digitaltechnik.

Aber jetzt will ich die beigelegten drei Listings ein wenig erleutern.

Das programm: FLAG?
dieses Programm zeigt den zustand sämtlicher 55flags grösser als 00an.
Will man nur die gesetzten Flags zur Anzeige bringen,so kann man nach Line 08 II einen GTO 0I-Befehl einfügen.Für das Anzeigen der gelöschten Flags ersetzt man einfach den GTO 02-Befehl in Line II durch einen GTO 0I_Befehel.
Line 25-27 können weggelassen werden,wenn in Line 08 för GTO 04 ein STOP-Befehl geschrieben wird.Sie sind mir presönlich jedoch angenehm.
damit der Alpha-Text bei FLAG.L: CLEAR nicht zu lang wird, sollte das Dezimalpunkt-Flag gelöscht sein,das ist aber für das Funktionieren des Programmes nicht unbedingt erforderlich.
Die Idee zu dieser Routine kam mir, als ich die SIZE-Routine von R.Berg, Kreuztal(siehe prisma I/4-80) ausprobierte.

Das Programm SI-KEY
Eigentlich braucht es keinen Kommentar zu dieser kleinen Routine.Der Grundgedanke war, die zehn normalen Töne des 4Iers mit jeweils nur einer einzigen Taste abrufen zu können(sonst mindestens zwei Tasten.)
Auf diese Art kann man bescheidene Kompositionen einfach zuwege bringen.

Das programm MATRIX
Das ist bereits ein etwas spezielleres Progamm.Es geht darum, an einer Ringmatritze(das ist ein verschleissteil von Pelletpressen in der Futtermittelindustrie) die Gesamtoberfläche, die Gesamtfläche aller Bohrlöcher auf dieser Oberfläche und das Verhältnis der beiden Flächen zueinander(Gesammtoberfläche entspricht I00%)zu errechnen.
Die Register 0I bis 05 werden für die Beschreibung der Matritze verwendet, die Register 06 bis 08 zur Ergebnisspeicherung.
Die Routine ist auf Komfort ausgelegt,sie liese sich leicht verkürzen.Es ist ein spezielles Anwenderprogramm für den Maschinenbau in der Futtermittelindustrie.
Vieleicht kann der eine oder andere eines dieser drei Programme verwerten.
In diesem Sinne

HAPPY PROGRAMMING

```
01 lbl FLAG?
   fix 0
   00.05501
   sto L
05 lbl 01
   isg L
   gto 03
   gto 04
09 lbl 03
   fs?ind L
   gto 02
   FLAG.
   arcl L
   ⊦: CLEAR
   aview
   pse
   gto 01
18 lbl 02
   FLAG.
   arcl L
   ⊦+ SET
   aview
   pse
   gto 01
25 lbl 04
   fix 5
   off
28 end
```

```
01 lbl SI-KEY
02 lbl 001
   TONE?
   aview
   pse
   sto L
   tone ind L
   gto 01
   end
```

```
01 lbl MATRIX
   fix 0
   DURCHM.=
   prompt
   sto 01
   GEL.BREITE=
   prompt
   sto 02
   LOCH/REIHE=
   prompt
   sto 03
   REIHEN=
   prompt
   sto 04
   LOCHRAD.=
   prompt
   sto 05
18 lbl 01
   rcl 01
   rcl 02
   *
   pi
   *
   100
   /
   sto 06
   TOT.FLACHE=
   arcl 06
   ⊦CM^2
   aview
   pse
32 lbl 02
   rxl 03
   rcl 04
   *
   rcl 05
   x^2
   *
   pi
   *
   100
   /
   sto 07
   TOT.LO.FLACHE=
   arcl 07
   ⊦CM^2
   aview
   pse
49 lbl 03
   rcl 07
   rcl 06
   /
   100
   *
   sto 08
   cla
   arcl 08
   ⊦%
   aview
   pse
   pse
   fix 5
   off
64 end
```

PROGRAMM MATRIX

User instruktions

prgm. eingeben
" starten Xeq MATRIX
variablen eingeben (nach jeder v. R/S drücken

Registerplan

00
01 Matritzeninnendurchmesser
02 gelochte Breite
03 Lochzahl pro Reihe
04 Anzahl Reihen
05 Lochradius
06 gesamte Oberfläche
07 " Lochfläche
08 Lochfläche % Oberfläche

Rüdiger Schultz

Magisches Quadrat

================

Andre Meyer-Lindenberg
Elliger Höhe 31
5300 Bonn 2 - Bad Godesberg
Telefon 0228/323255

HP-41C ohne Erweiterungen
Size: 017 (7 verwendete Speicher)
Assignments (User): MAGQ

Ein "magisches Quadrat" entsteht, wenn man in einem n·n-Quadrat alle Zahlen von 1 bis n^2 so anordnen kann, daß die Summe der Zahlen in jeder Zeile, Spalte und Diagonale gleich ist. Die Aufgabe der Berechnung dieser Gebilde ist eines der ältesten (die erste Lösung findet sich schon 2000 v. Chr. in China) und faszinierendsten Gebiete der Unterhaltungsmathematik. So ziemlich alles, was in der Mathematik Rang und Namen hat, hat sich mit diesem Thema befaßt; viele dicke Wälzer sind allein über dieses Thema geschrieben worden, so viele, daß eine Bibliographie dieser Werke selber schon wieder ein dickes Buch wäre.

Die Methode, die dieses Programm verwendet, stammt aus dem 13/14. Jh. von einem Inder namens Moschopulos. Nach dem ersten Europäer, der darüber (im 19. Jh.) berichtete, nennt man sie auch manchmal die "de la Loubere"-Methode. Auf die genaue Angabe der Methode, sowie auf deren Beweis möchte ich hier verzichten; sie mußte ohnehin völlig neugefaßt und mathematisch aufgeknackt werden, um eine PTR-Implementierung sinnvoll zu machen.

Mit Hilfe dieser Methode kann der HP-41C nun schnell und sicher ein magisches Quadrat beliebiger Seitenlänge errechnen - wenn die Seitenlänge ungerade ist! Bei geraden Seitenlängen versagt die Methode (genauso wie die Methoden für gerade Quadrate bei ungeraden versagen; allgemeine Methoden gibt's (noch?) nicht) leider, man braucht dafür ein anderes Programm.

Die Handhabung des Programms ist relativ simpel: Start mit "MAGQ". Der Rechner fragt nach der "Größe" (= Seitenlänge) des Quadrates. Geben Sie die gewünschte Seitenlänge ein und starten Sie den Rechner wieder. Der Rechner zeigt nun kurz die Summe in jeder Zeile, Spalte und Diagonale an (∑(N):) und gibt dann das magische Quadrat spaltenweise aus. Vor Beginn jeder neuen Spalte ertönt ein Piepser. Jedes Element steht für ca. 2 sek. in der Anzeige - wem das zu kurz ist, der ersetze die Schritte 49/5Ø durch STOP. Nach der Ausgabe des letzten Elements stoppt der Rechner mit BEEP und der Seitenlänge des Quadrates in der Anzeige. Durch Druck auf die ALPHA-Taste kann die Quadratsumme wiederum abgerufen werden.

Literatur (für Interessierte) kann ich gerne beschaffen bzw. (gegen Unkostenerstattung) kopieren. Ansonsten wünsche ich viel Spaß mit dem Programm und bin

HAPPY PROGRAMMING

Magisches Quadrat
================

HP-41C ohne Erweiterungen
Andreas Meyer-Lindenberg (258), 12.01.81

```
Ø1 LBL "MAGQ"
Ø2 CLΣ
Ø3 "GROESSE ?"
Ø4 TONE 9
Ø5 PROMPT
Ø6 Σ+
Ø7 LAST X
Ø8 STO 1Ø
Ø9 STO 13
1Ø STO 14
11 ST+ 16
12 2
13 ST+ 16
14 ST/ 16
15 ST/ 12
16 ST+ 12
17 /
18 ST+ 12
19 STO 15
2Ø FRC
21 X=Ø?
22 1/X
23 ST+ 15
24 RCL 13
25 ENTER↑
26 X↑2
27 1
28 ST- 14
29 +
3Ø *
31 2
32 /
33 "Σ(N):"
34 ARCL X
35 AVIEW
36 LBL Ø2
37 TONE 9
38 DSE 15
39 GTO Ø1
4Ø STO 15
41 STO 14
42 1
43 STO 12
44 STO 16
45 LBL Ø1
46 RCL 12
47 LBL Ø3
48 CLD
49 PSE
5Ø PSE
51 STO Ø9
52 RCL 15
53 RCL 13
54 1
55 +
56 RCL 11
57 -
58 X≠Y?
59 GTO Ø4
6Ø RCL 16
61 GTO Ø5
62 LBL Ø4
63 RCL 14
64 -
65 Ø
66 X≠Y?
67 RCL 13
68 1
69 +
7Ø RCL Ø9
71 +
72 LBL Ø5
73 DSE 11
74 GTO Ø3
75 2
76 ST+12
77 ST-14
78 1
79 ST+ 16
8Ø RCL 13
81 ST+ 12
82 STO 11
83 DSE 1Ø
84 GTO Ø2
85 BEEP
86 END
```


Andreas Meyer-Lindenberg
Elliger Höhe 31
5300 Bonn 2 - Bad Godesberg
Telefon 0228/323255

Liebe Mitglieder!

Wie Ihr aus der letzten Prisma wißt, werde ich mich ab sofort um die synthetische Programmierung in unserem Club kümmern. Ich habe vor hier einiges stark zu verändern. Grund dafür ist das Buch "Synthetic Programming on the HP-41C" von Dr. W. C. Wickes. Es ist erschienen bei

Larken Publications
P.O. Box 967
College Park, MD 2o74o U.S.A.

zum Preis von US$ 11,5o incl. aller Unkosten. Wer das Geld in bar mit der Bestellung verschickt, kann in 4-6 Wochen mit seinem Exemplar rechnen. Ich erwähne dies so ausführlich, weil ich das Buch zur Grundlage der Rubrik "Synthetische Programmierung" mache. Jeder, der sich fur dieses Thema interessiert, sollte das Grundwissen, das dieses hervorragende Buch vermittelt besitzen. Es ist als (sogar von HP empfohlenes) Nachschlagewerk unerläßlich. Es werden von mir nur noch solche Artikel über synthetische Programmierung zum Abdruck gebracht, die über, das in dem Buch gesetzte Level hinaus wirklich neue Informationen bieten. Neben dieser inhaltlichen Einschränkung, habe ich noch eine zur Form von eingesandten Artikeln. Schreibt bitte alles, von dem Ihr glaubt, daß es für alle interessant ist, auf DIN A4-Blätter mit einer Schreibmaschiene und laßt an allen Seiten genug Rand. Bitte keine beidseitig beschrifteten Blätter. Ich sehe meine Aufgabe nicht als Clubsekretärin und werde Beiträge, die diese Bedingung nicht erfüllen zwar dankbar zur Kentnise nehmen, sie aber nicht an Oliver weiterleiten. Ansonsten freue ich mich über jede Post von Euch und bemühe mich (selbstadressierten und frankierten Rückumschlag nicht vergessen!) eventuelle Fragen ausreichend zu beantworten. Happy (synthetic) programming!

Rolf

Rolf Wach (75)
Thorwaldsenstr. 5
6o9o Rüsselsheim

Als Alternative zu den Wandlern vom Typ wie in der Standard-Programmsammlung, die das Dezimalequivalent der einzelnen Hexdigits durch Vergleich ermitteln, bietet sich mit der "synthetischen" Methode die Möglichkeit der direkten Umwandlung. Zudem wird kein weiterer Datenspeicher außer Stack und Alpha benötigt.

Benutzung der beiden Routinen:

LBL"Z" wandelt die bis zu achtstellige Hexzahl in Alpha ins Dezimalsystem um, löscht Alpha, zerstört den Stack, übergibt das Ergebnis im x-Register. Zeile 7:STR(127,Ø,Ø,Ø,Ø,2,2)

```
 01•LBL "Z"
 02 1
 03 CHS
 04 0
 05•LBL 10
 06 RCL d
 07 "⊢◆◆◆◆x̄x̄
"
 08 0
 09 X<> \
 10 ASTO d
 11 CF 03
 12 STO [
 13 CLX
 14 X<>Y
 15 X<> d
 16 +
 17 X=0?
 18 GTO 09
 19 40
 20 -
 21 X>0?
 22 DSE X
 23 ""
 24 10
 25 +
 26 16
 27 R↑
 28 ISG X
 29 ""
 30 Y↑X
 31 LASTX
 32 RDN
 33 *
 34 +
 35 GTO 10
 36•LBL 09
 37 RDN
 38 .END.
        CAT 1
LBL'Z
END
64 BYTES
```

LBL"HEX" wandelt den 65536er Rest der Zahl im X-Register in eine vierstellige Hexadezimalzahl um, die in Alpha übergeben wird. Der Stack wird gelöscht.

Zeile 14: 24Ø (=FØ) NOP

BSP:

Was ist $1Ø752_{1Ø}$ in Hex?

1Ø752
XEQ"HEX"
2AØØ

Zurückwandeln:

XEQ"Z"
1Ø752 .

<u>WER</u>

schreibt eine "HEX"-Routine für acht Stellen? Doch bitte ohne Size!!

Hexy Programming

wünscht

Sven

```
 01•LBL "HEX
"
 02 CLA
 03 4
 04 STO \
 05 X<>Y
 06•LBL 02
 07 STO [
 08 16
 09 MOD
 10 9
 11 -
 12 X>0?
 13 ISG X
 14 ""
 15 39
 16 +
 17 1 E2
 18 ST/ ]
 19 /
 20 ST+ ]
 21 RCL [
 22 16
 23 /
 24 INT
 25 DSE \
 26 GTO 02
 27 RCL ]
 28 1
 29 +
 30 CLA
 31 STO [
 32 ASTO [
 33 ASTO X
 34 "ABCD"
 35 ARCL X
 36 ASHF
 37 CLST
 38 AVIEW
 39 .END.
        CAT 1
LBL'HEX
.END.
70 BYTES
```

Frank Alten...

aten und Rechnen lassen

eder, der schon einmal die Zeitschrift "Stern" gelesen hat, ird über kurz oder lang auf die Rubrik "Raten und Rechnen" estoßen sein. Eines dieser Rätsel -zugegebenermaßen eines er leichteren Kategorie(wegen der Rechenzeit)- habe ich mir erausgegriffen um an Ihm zu zeigen, wie man mit einem Rech- er und ein paar Überlegungen derartige Probleme lösen kann.

ufgabe:



Raten und Rechnen

Jedes Karo bedeutet eine Ziffer, gleiche Ziffern haben gleiche Karos. Durch Probieren, Nachdenken und Rechnen ist die Aufgabe durch Aufschreiben der richtig gefundenen Zahlen an Stelle der Karos waagerecht und senkrecht lösbar.

ls ersten Schritt bezeichne man zweckmäßigerweise die "Karos" it Buchstaben oder gleich mit Registernamen um von der unüber- ichtlichen Mengenlehre-Schreibweise auf einen klaren algebra- schen Ausdruck zu kommen.
ies ergibt:

ABC	-	DE	=	ACF
+		*		-
ABF	-	DA	=	AGC
DHEA	:	EDB	=	A

ies sind sechs Gleichungssysteme mit 10 Unbekannten und somit uf konventionelle Weise nicht zu lösen. Man kann derartige leichungssysteme aber immer derart reduzieren, daß eine Gleich- ng mit vier oder fünf Variablen übrig bleibt:

sp.1: rechte, senkrechte Spalte:
100A+10C+F-(100A+10G+C)=A
100A+10C+F-100A-10G-C =A
9C+F-10G=A

sp.2: mittlere, senkrechte Spalte:
Eine Umformung ist nicht nötig, da insgesmt schon vier Variable vorliegen.

a jedes der Karos(jetzt Buchstaben) zehn verschiedene Zustände [illegible]

der Begle[illegible], während man bei [illegible] Kombinationen nur ca. 3,5 Stunden benötigt.

Ein zweites Kriterium über der Auswahl eines Gleichungssystems ist, daß im Ergebnisse möglichst eine der Variablen aus den Operanden enthalten ist, da sich dadurch die Anzahl der Auswahllösungen deutlich verringert. Dies ist -um bei den Beispielen zu bleiben- beim zweiten Term gegeben, nicht aber beim ersten, da dort vier unabhängige Variable existieren. Es wird also DExDA=EDB zur weiteren Bearbeitung herangezogen.
Man braucht nun nur für die Buchstaben D, E, A und B die Ziffern Ø bis 9 einzusetzen und alle Mäglichkeiten durchzuspielen. Dazu verwenden wir den Rechner.

Dabei gibt es zwei recht akzeptable Wege:

1. Das Quadtupel DEAB ist eine ganze Zahl zwichen ØØØØ und 9999. Man braucht also nur von ØØØØ nach 9999 hochzuzählen, die einzelnen Elemente(jede Stelle dieser Zahl entspricht einem Zustand einer der Variablen) zu isolieren und in die Gleichung DExDA=EDB einzusetzen. Ist das Produkt der linken Seite tatsächlich mit der Zusammensetzung der Elemente auf der rechten Seite identisc so kann angenommen werden, daß es sich um eine Teillösung handelt(Weitere Auswahl wird später getroffen)! Das Verfahren ist zwar recht übersichtlich, hat aber den Nachteil, daß die Isolierung der einzelnen Variablen aus der ganzen Zahl recht lange dauert und sich die Rechenzeit entsprechend verlängert.

2. Das zweite Verfahren funktioniert ähnlich, nur wird hier nicht von einer ganzen Zahl ausgegangen. Für jeden Buchstaben wird ein Datenspeicher benötigt. Der erste Buchstabe wird nun von Ø bis 9 hochgezählt. Ist 9 erreicht, wird der nächte Buchstabe um eins hochgezählt und der erste auf Ø gesetzt usw. Das Verfahren wird solange fortgesetzt, bis alle Möglichkeiten durchexerziert sind. Die einzelnen Zustandsformen der Buchstaben wird in obige Gleichung eingesetzt und überprüft, ob es zu einer sog. "wahren Aussage" kommt. Ist dies der Fall, so hat man eine der Teillösungen gefunden. Diese Methode hat zwar den Nachteil, daß Sie recht unübersichtlich ist, dafür aber relativ wenig Rechenzeit beansprucht.
   Aus diesem Grund soll darauf näher eingegangen werden.

Veranschaulicht ist ein Programm nach Verfahren 2. in Fig.1 (als Struktogramm) und realisiert in Fig.2(Progamm für HP-41C). Zu Starten ist das Prgm. mit (XEQ'RATEN). Man erhält dann folgende Ausgaben:



Von diesen fünfzehn Lösungen erfüllt nur eine die Bedingung, daß verschiedene Buchstaben auch verschiedene Ziffern darstellen müssen. Dies ist 1286 (D=1; E=2; A=8; B=6).
Zur weiteren Ermittlung von anderen Veränderlichen, zieht man nun das Gleichungssystem DHEA:EDB=A heran, daß außer den nun bekannten Variablen D, E, A und B noch die unbekannte Variable H enthält. Zur Vereinfachung der Aufgabe wird diesen wie folgt umgeformt:

aus DHEA:EDB=A
wird: DHEA=A*EDB

Die bekannten Veränderlichen werden eingesetzt:

1H28=8*216

Also muß H=7 sein.
Als Rest-Lösungsmenge bleibt: L=(3,4,5,9,∅)

Die nächsten Variablen, die wir aufgreifen wollen, seien C und F. Dazu wird in das System ABC+ABF=DHEA die bekannten Werte eingesetzt:

```
  86C
 +86F
 1728
```

Es ist ersichtlich, daß C+F=8 sein muß. In der oben angegebenen Lösungsmenge sind nur die Ziffern 3 und 5, die zusammen acht ergeben. Aber welcher Buchstabe entspricht welcher Ziffer?
Dazu nehme man das Gleichungssystem ABC-DE=ACF und verfahre wie oben:

```
  86C
 - 12
  8CF
```

Durch probieren erhält man: C=5; F=3.
Die Lösungsmenge hat sich verringert: L_1=(∅,4,9)

Es gilt jetzt noch, den Ziffernwert für die Variable G zu erhalten:

ACF-AGC=A
bzw.:
853-8G5=8

Also: G=4.

Das komplette System sieht dann so aus:

```
 865 -  12 = 853
  +     *     -
 853 -  18 = 845
1728 : 216 =   8
```

Viel Spaß und
Happy Programming

Raten und Rechnen lassen

Fig.1

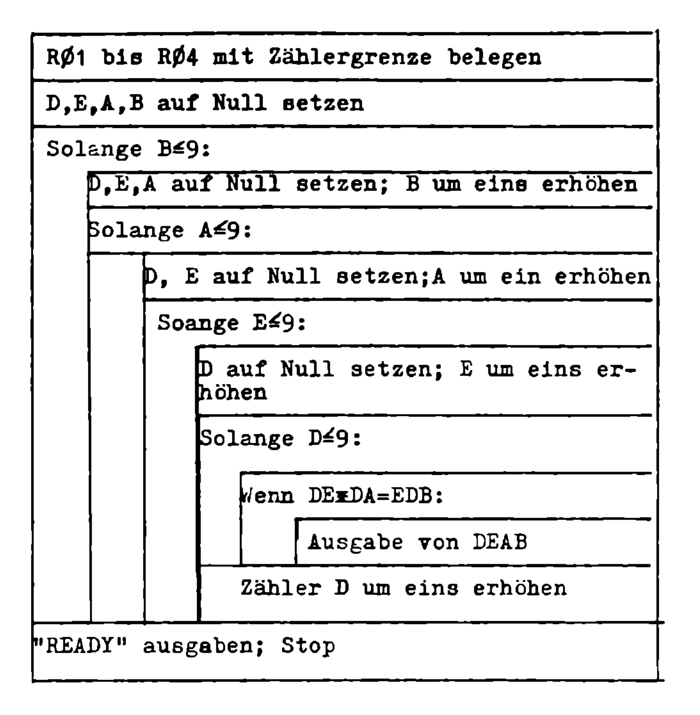
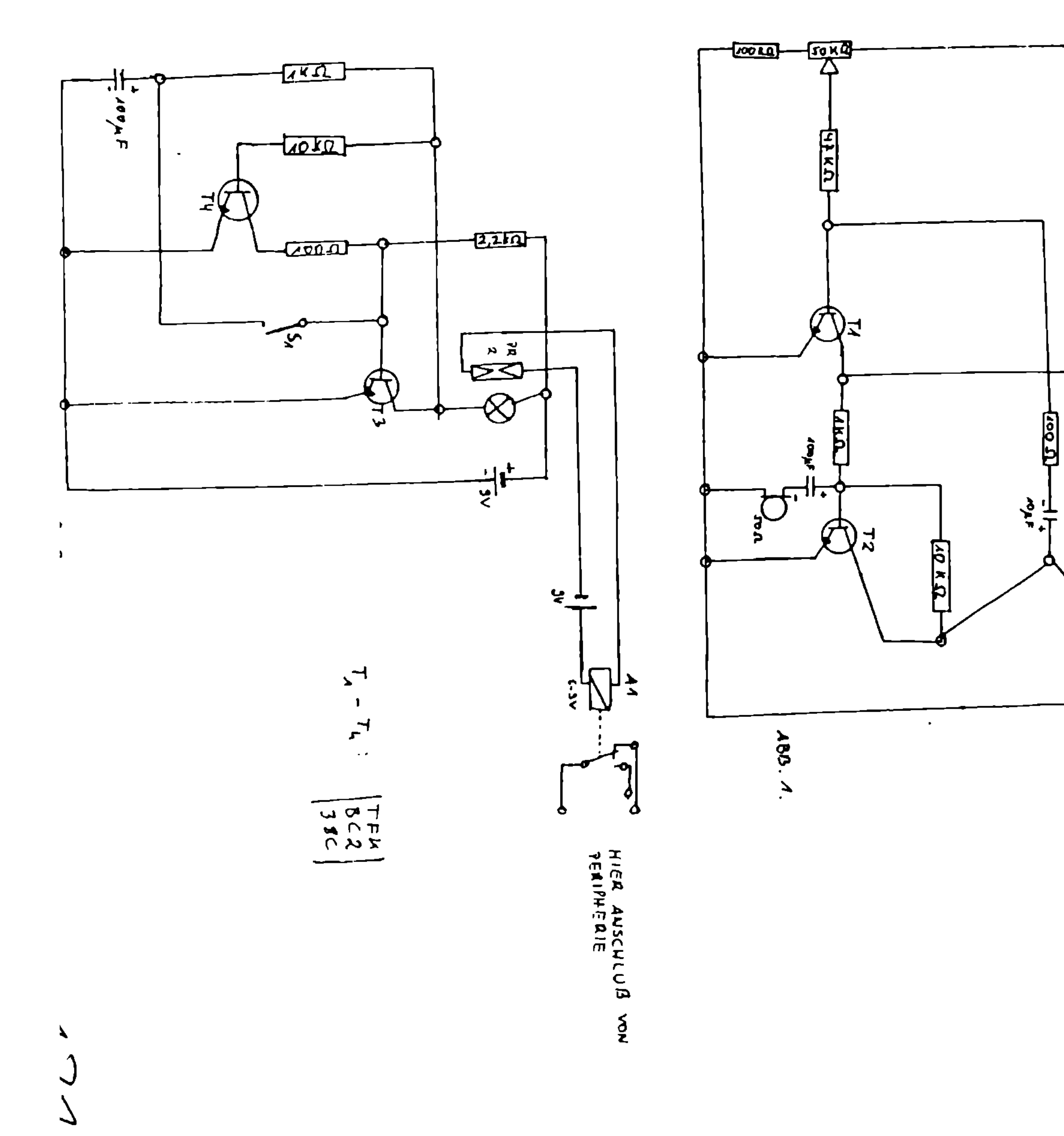


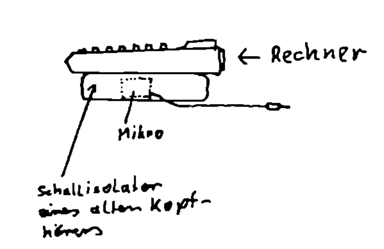
Fig.2

```
01 LBL'RATEN
02 ,ØØ9
03 STO Ø1
04 STO Ø2
05 STO Ø3
06 STO Ø4
07 LBL ØØ
08 RCL Ø1
09 INT
10 1Ø
11 *
12 RCL Ø2
13 INT
14 +
15 RCL Ø1
16 INT
17 1Ø
18 *
19 RCL Ø3
20 INT
21  +
22  *
23 STO ØØ
24 RCL Ø2
25 INT
26 1ØØ
27 *
28 RCL Ø1
29 INT
30 1Ø
31 *
32 +
33 RCL Ø4
34 INT
35 +
36 RCL ØØ
37 X=Y?
38 GTO Ø1
39 LBL Ø2
40 ISG Ø1
41 GTO ØØ
42 ,ØØ9
43 STO Ø1
44 ISG Ø2
45 GTO ØØ
46 STO Ø2
47 ISG Ø3
48 GTO ØØ
49 STO Ø3
50 ISG Ø4
51 GTO ØØ
52 'READY
53 PROMPT
54 LBL Ø1
55 TONE 9
56 FIX Ø
57 'DEAB=
58 RCL Ø1
59 INT
60 ARCL X
61 RCL Ø2
62 INT
63 ARCL X
64 RCL Ø3
65 INT
66 ARCL X
67 RCL Ø4
68 INT
69 ARCL X
70 AVIEW
71 GTO Ø2
72 .END.
```

Akustisches Schalt-Interface für den HP-41 C

Ein Wort vorweg: Ich bin kein professioneller Elektroniker/ Bastler. Bei der Entwicklung dieses Interface wurde auf einen Elektronik-Baukasten zurückgegriffen. Dabei habe Ich besonderen Wert darauf gelegt, daß es funktioniert und kein Eingriff im Rechner erforderlich ist. Es ist bestimmt in vielen Punkten zu optimieren; eventuell ist sogar eine andere Konfiguration von Vorteil. Ich möchte mit dem Interface auch keine "Fertiglösung" anbieten, sondern vielmehr zu weiterentwicklungen anregen bzw. zeigen, daß man auch als Laie mit recht primitiven Mitteln ein Interface o.ä. schaffen kann. Doch zur Sache! Abbildung 1. zeigt das eigentliche Gerät. Man lege das Mikrofon unter oder neben den Rechner; stellt den Potentiometer so ein, daß das Lämpchen bei ertönen des Tones 9 (TONE 9) aufblinkt. Bei jedem weiteren ertönen dieses Tones, schaltet nun das Relais A2 durch (für ca 1,5sek.). Zum permanenten einschalten eines Gerätes, schließe man die im Schaltplan mit "Pfeil" gekennzeichneten Ausgänge des Relais A2 anstelle des Schalters S1 im Abb. 2 an. Ertönt nun TONE 9, so schaltet sich das Gerät, das an Relais A1 angeschlossen ist ein. Ertönt der Ton zum zweiten mal, wird das Gerät wieder ausgeschaltet.
Man kann nun z.B. einen Spielzeugroboter oder eine Elektronische Uhr steuern. Besonders die Uhr ist interesant, da man damit z.B. die exakte Laufzeit eines Prgm.s ermitteln kann, ohne dabei sein zu müßen (besonders bei rel. langen Prgm.s)!
Um das einschalten des Interface durch Umweltgeräuche zu verhindern, empfihlt es sich, das Mikrofon unter den Rechner zu legen. Außerdem kann man das Mikro zusätzlich durch den Gehörschutz eines alten Kopfhörers isolieren(siehe Zeichnung).





Diese Isolatoren haben den Vorteil, daß sie ziemlich exakt die Größe des Rechners aufweisen und außerdem einen zusätzlichen Resonanzraum unter dem Rechner schaffen. Alles andere: selbst Ausprobieren!!!!

Euer Frank

Frank Altensen
Uhlandstr. 9
6365 Rosbach 3

P.S. Schreibt mir doch mal eure Erfahrungen oder Verbesserungsvorschläge. Ich würde mich freuen!

INTEGRAL NACH WEDDLE *

```
01♦LBL "AB"
02 SF 25
03 XEQ ""
04 "               3→♦"
05♦LBL A
06 "F ?"
07 AON
08 STOP
09 ASTO 00
10 AOFF
11♦LBL 07
12 CF 22
13 "<a,b> ?"
14 PROMPT
15 FC?C 22
16 GTO A
17 SF 25
18 XEQ ""
19 "uA♦ãCv♦ũ00ã"
20 STO 01
21♦LBL 08
22 SF 25
23 GTO "ᵀ"
24 "'uv!hB@        0
"
25 DEG
26 GTO IND 01
27♦LBL 00
28♦LBL 06
29 41
30 GTO 10
31♦LBL 05
32♦LBL 01
33 216
34 GTO 10
35♦LBL 02
36♦LBL 04
37 27
38 GTO 10
39♦LBL 03
40 272
41 GTO "A"
42♦LBL 10
43 SF 25
44 GTO ""
45 "Bw×"
46 GTO 08
47 RCL ]
48 840
49 /
50 RCL \
51 6
52 *
53 *
54 TONE 9
55 STOP
56 GTO 07
57 END
```

DECODIERTE VERSION, SVEN

```
01♦LBL "AB"
02 FS? 51
03 STO c
04 ADV
05♦LBL A
06 "F ?"
07 AON
08 STOP
09 ASTO 00
10 AOFF
11♦LBL 07
12 CF 22
13 "<a,b> ?
"
14 PROMPT
15 FC?C 22
16 GTO A
17 STO IND
c
18 STO [
19 -
20 6
21 /
22 STO \
23 .006
24 STO 01
25♦LBL 08
26 RCL [
27 RCL \
28 RCL 01
29 INT
30 *
31 +
32 SF 25
33 XEQ IND
00
34 GTO IND
01
35♦LBL 00
36♦LBL 06
37 41
38 GTO 10
39♦LBL 05
40♦LBL 01
41 216
42 GTO 10
43♦LBL 02
44♦LBL 04
45 27
46 GTO 10
47♦LBL 03
48 272
49 GTO "A"   ←┐
50♦LBL 10      │
51 *           │
52 ST+ ]       │
53 ISG 01     ─┘
```

Zur Anwendung des PBJ

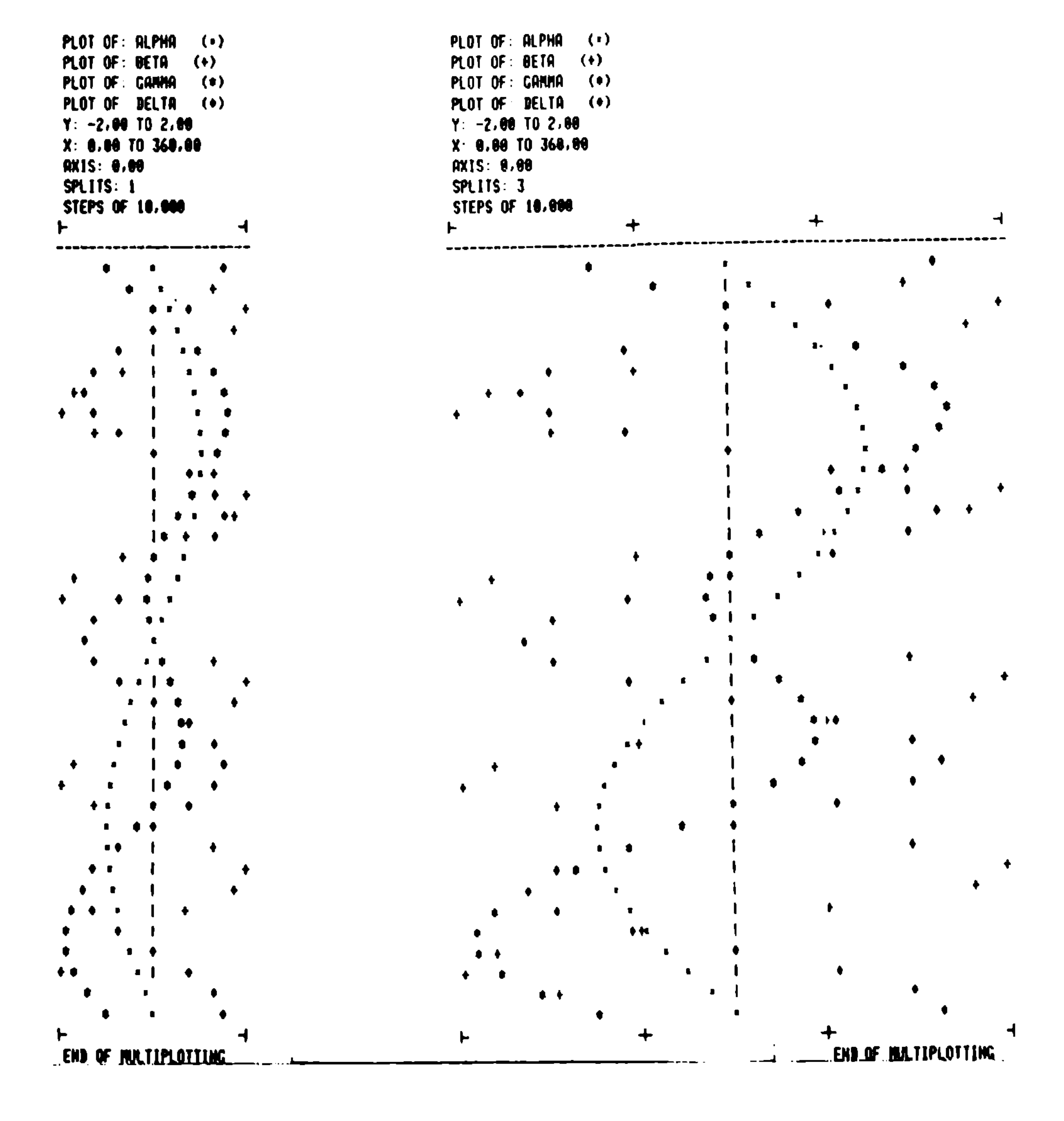
ein Beispielprogramm von Michael Fehse :
Seit der Beseitigung des PRIVATE schien das "geschützte" Programm ausgestorben. Michael fand einen Ausweg. Er ließ sein Programm unter der "Tarnkappe" der Textbytes zum Alphatext werden. Diese Textbytes werden mittels PBJ übersprungen, sobald das Programm ausgeführt wird.
Links die so getarnte Version (numerisches Integral nach Weddle), rechts die von mir wieder normalisierte.
Bnutzung:(F programmiert)

```
                 XEQ "AB"
F ?
X                RUN
<a,b> ?
           3.0000 ENTER↑
           0.0000    RUN
                  VIEW X
                  4.5000
```

$\int_0^3 x\,dx = 4{,}5$

```
54 GTO 08
55 RCL ]
56 840
57 /
58 RCL \
59 6
60 *
61 *
62 TONE 9
63 STOP
64 GTO 07
65 END
```

Sprung über LbL 10
[kein →...]



Ich habe in Robert Klauc's Multiplotter-Programm in folgenden Punkten abgeändert:

- automatischer Ausdruck der einzelnen Druckstreifen
- Die Anzahl der Druckstreifen ist von minimal Einem bis unendlich Vielen erlaubt
- Die Druckzeit für die obenstehenden Beispiele betrug ca. 15 min pro Streifen
- Beim Test der zulässigen Anzahl der Funktionen (am Anfang des Programms) wurde bei Robert Klauc's Programm die Null nicht überwacht
- Das zu plottende Unterprogramm wurde bei Robert Klauc zuerst im Speicherregister 21 abgelegt und erst dann wurde indirekt zum Unterprogramm gesprungen, was Zeit- und Speicherplatzverlust bedeutet. Ich habe es soweit abgeändert, dass das Unterprogramm nun indirekt über das Y-Register aufgerufen wird.
- Der minimale SIZE hat sich dadurch auf 28 erhöht
- Das Programm belegt 106 Programmregister (736 Bytes)
- Es sind min. 2 RAM nötig
- Die Bedienung des Programms ist noch gleich, wie bei Robert Klauc

Fred Huber

```
01♦LBL "PLO
T"
02 CF 29
03 FIX 0
04 4
05 "NO. OF
F<X>?"
06 PROMPT
07 X>0?
08 X>Y?
09 GTO "PLO
T"
10 1 E3
11 /
12 1
13 +
14 STO 00
15 AON
16♦LBL 15
17 "NAME F"
18 ARCL X
19 "├ ?"
20 PROMPT
21 ASTO IND
X
22 ISG X
23 GTO 15
24 AOFF
25♦LBL "NEW
"
26 "SPLITS
?"
27 PROMPT
28 STO 26
29 X<=0?
30 GTO "NEW
"
31♦LBL 16
32 "Y MIN?"
33 PROMPT
34 STO 05
35 "Y MAX?"
36 PROMPT
37 STO 06
38 RCL 05
39 X>Y?
40 GTO 16
41 -
42 RCL 26
43 /
44 STO 27
45 "AXIS?"
46 PROMPT
47 STO 25
48♦LBL 17
49 "X MIN?"
50 PROMPT
51 STO 07
52 "X MAX?"
53 PROMPT
54 STO 08
55 RCL 07
56 X>Y?
57 GTO 17
58 "X INC?"
59 PROMPT
60 X<0?
61 XEQ 18
62 STO 09
63 CF 12
64 1
65 STO 11
66 43
67 STO 12
68 42
69 STO 13
70 10
71 STO 14
72 RCL 00
73 STO 10
74 ADV
75♦LBL 19
76 " PLOT O
F: "
77 ARCL IND
10
78 "├  "
79 ACA
80 40
81 ACCHR
82 RCL 10
83 10
84 +
85 RCL IND
X
86 ACCHR
87 41
88 ACCHR
89 PRBUF
90 ISG 10
91 GTO 19
92 FIX 2
93 " Y: "
94 ARCL 05
95 "├ TO "
96 ARCL 06
97 PRA
98 " X: "
99 ARCL 07
100 "├ TO "
101 ARCL 08
102 PRA
103 " AXIS:
"
104 ARCL 25
105 PRA
106 FIX 0
107 " SPLITS
: "
108 ARCL 26
109 PRA
110 FIX 3
111 " STEPS
OF "
112 ARCL 09
113 PRA
114 RCL 05
115 STO 06
116♦LBL 20
117 RCL 06
118 STO 05
119 RCL 27
120 ST+ 06
121 XEQ 21
122 "-------"
123 ASTO L
124 ARCL L
125 ARCL L
126 ARCL L
127 PRA
128 FIX 0
129 RCL 06
130 RCL 05
131 -
132 161
133 /
134 STO 23
135 RCL 05
136 CHS
137 RCL 25
138 +
139 RCL 23
140 /
141 RND
142 X<0?
143 200
144 STO 16
145 ,124
146 +
147 STO 24
148 RCL 07
149 STO 15
150♦LBL 22
151 RCL 24
152 STO 16
153 RCL 00
154 STO 10
155♦LBL 23
156 RCL IND
10
157 RCL 15
158 XEQ IND
Y
159 RCL 05
160 -
161 RCL 23
162 /
163 RND
164 X<0?
165 200
166 RCL 10
167 10
168 +
169 X<>Y
170 RCL IND
Y
171 1 E3
172 /
173 +
```

```
174 RCL 10
175 16
176 +
177 X<>Y
178 STO IND
Y
179 ISG 10
180 GTO 23
181 RCL 00
182 16.016
183 +
184 STO 21
185◆LBL 24
186 RCL 21
187 STO 22
188 1
189 -
190 RCL IND
X
191◆LBL 25
192 RCL IND
22
193 X<Y?
194 X<>Y
195 STO IND
22
196 RDN
197 ISG 22
198 GTO 25
199 STO IND
Y
200 ISG 21
201 GTO 24
202 RCL 16
203 FRC
204 1 E3
205 *
206 STO 22
207 124
208 X=Y?
209 XEQ 26
210 161
211 RCL 16
212 X>Y?
213 GTO 00
214 SKPCOL
215 RCL 22
216 ACCHR
217◆LBL 27
218 RCL 00
219 16.016
220 +
221 STO 21
222◆LBL 28
223 RCL IND
21
224 FRC
225 1 E3
226 *
227 STO 22
228 124
229 X=Y?
230 XEQ 01
231 161
232 RCL IND
21
233 INT
234 X>Y?
235 GTO 00
236 RCL 21
237 1
238 -
239 X<>Y
240 RCL IND
Y
241 INT
242 -
243 7
244 FS?C 05
245 CLX
246 -
247 X<0?
248 GTO 02
249 SKPCOL
250 RCL 22
251 ACCHR
252◆LBL 29
253 ISG 21
254 GTO 28
255◆LBL 00
256 PRBUF
257 RCL 09
258 ST+ 15
259 RCL 08
260 RCL 15
261 X<=Y?
262 GTO 22
263 XEQ 21
264 DSE 26
265 GTO 20
266 BEEP
267 "END OF
"
268 "⊢MULTIP
LOTTING"
269 PRA
270 FIX 2
271 ADV
272 ADV
273 ADV
274 ADV
275 STOP
276◆LBL 01
277 RCL 21
278 INT
279 LASTX
280 FRC
281 1 E3
282 *
283 X=Y?
284 RTN
285 1
286 ST+ Z
287 RCL IND
Z
288 RCL IND
21
289 -
290 7
291 -
292 0
293 X<=Y?
294 RTN
295◆LBL 02
296 RCL 21
297 1
298 -
299 RCL IND
X
300 STO IND
21
301 GTO 29
302◆LBL 26
303 RCL 17
304 RCL 16
305 -
306 7
307 -
308 0
309 X<=Y?
310 RTN
311 STO 16
312 SF 05
313 GTO 27
314◆LBL 18
315 ABS
316 RCL 08
317 RCL 07
318 -
319 X<>Y
320 /
321 RTN
322◆LBL 21
323 3
324 SKPCOL
325 127
326 ACCOL
327 8
328 ACCOL
329 ACCOL
330 ACCOL
331 ACCOL
332 ACCOL
333 ACCOL
334 148
335 SKPCOL
336 8
337 ACCOL
338 ACCOL
339 ACCOL
340 ACCOL
341 ACCOL
342 ACCOL
343 127
344 ACCOL
345 PRBUF
346 END
```

Hier habe ich für den HP-41C einige kleinen Programme:

1. Programm "XROM": Es ist als Ergänzung zum Key-Assignment-PRGM gedacht; Um Funktionen von Zubehörteilen zu speichern, benutzt der Rechner die XROM-Nummern, die aus 2 Bytes bestehen: Die ersten 5 Bits lassen den Rechner erkennen, daß es sich um XROM handelt, und die übrigen 3 Bits sowie die 8 Bits des nächsten Bytes enthalten die Information über Nummer des Zubehörteils und Nummer der Funktion. Es ist z.B. bekannt, daß PRSTK die Mezeichnung XROM 29,19. Das Programm rechnet nun aus das von den Funktionsbytes 160-167 das letzte in Frage kommt, und die Adresse des Befehls 83 lautet. Ins Key-Ass. PRGM muß also 167 ENTER 83 ENTER (Taste) R/S eingegebem werden muß.

2. Programm "STAT": Nach den vielen SIZE-Programmen hier eines, daß den Statistik-Register-Block lokalisiert. Nach dem Versuch, diese Information aus dem Register c mit Hilfe des Registers d zu erhalten, gab ich das nach kurzer Zeit auf, da der Rechner begann, sich selbst mit der Ziffer 1 zu programmieren ! Hier also eine konventionelle Version. Mit XEQ "STAT" starten, Nummer des 1. Registers in X.

3. Programm "GET": Um einen Buchstaben aus einem bestimmten Register ins ALPHA-Register zu holen. Nur das Alpha-Register wird verändert.

4. Programm "GETA": Soll das gleiche Problem wie 3 lösen, das Zeichen steht diesmal im Alpharegister selbst. Bei 3 Register- und Zeichen-Nummer eingeben, hier nur die Nummer des Zeichens.

5. Programm "GAMMA": Berechnet die Gammafunktion; Argument eingeben, RTN R/S oder XEQ A oder XEQ "GAMMA".

Ralf Pfeifer
Rubensstr. 5
5000 Köln - 50


```
01♦LBL "XROM"
02♦LBL A
03 ENTER↑
04 FRC
05 1 E2
06 *
07 STO 00
08 RDN
09 INT
10 ENTER↑
11 ENTER↑
12 4
13 MOD
14 -
15 LASTX
16 64
17 *
18 ST+ 00
19 RDN
20 4
21 /
22 160
23 +
24 RCL 00
25 1 E3
26 /
27 +
28 .END.
```

```
01♦LBL "STAT"
02♦LBL A
03 5
04♦LBL 00
05 RCL IND X
06 0
07 ENTER↑
08 Σ+
09 RDN
10 RDN
11 RCL IND Y
12 X≠Y?
13 GTO 01
14 RCL Z
15 1
16 +
17 GTO 00
18♦LBL 01
19 -4
20 R↑
21 +
22 ST- IND L
23 1
24 -
25 .END.
```

```
01♦LBL "GET"
02 "REG ?"
03 PROMPT
04 "NR ?"
05 PROMPT
06 5
07 -
08 CLA
09♦LBL 00
10 X<=0?
11 "├ "
12 ISG X
13 GTO 00
14 ARCL IND Y
15 ASTO Y
16 " "
17 ARCL Y
18 ASHF
19 AVIEW
20 .END.
```

```
01♦LBL "GETA"
02♦LBL A
03 1
04 -
05 ENTER↑
06 ENTER↑
07 6
08 /
09 INT
10♦LBL 00
11 X≠0?
12 ASHF
13 DSE X
14 GTO 00
15 CLX
16 6
17 MOD
18 5
19 X<>Y
20 -
21 ASTO Y
22 CLA
23♦LBL 01
24 X≠0?
25 "├ "
26 DSE X
27 GTO 01
28 ARCL Y
29 ASTO Y
30 " "
31 ARCL Y
32 ASHF
```

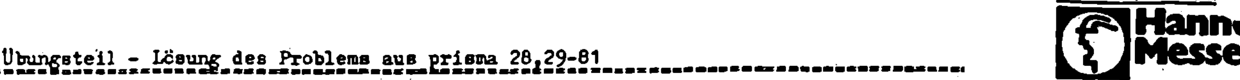
```
PRP "GAMMA"

01♦LBL "GAMMA"
02♦LBL A
INT  LASTX  X≠Y?  GTO 00
1  -  FACT  RTN

11♦LBL 00
CF 00  1  STO 01  X<>Y
STO 00  X<0?  SF 00
X<0?  -  5  X<>Y

23♦LBL 01
X>Y?  GTO 02  ST* 01  1
+  GTO 01

30♦LBL 02
ENTER↑  X↑2  RCL X  742
*  602  +  *  13  +  18
*  X<>Y  X↑2  1113  *
865,9  +  X<>Y  /  1  -
X<>Y  LN  +  *  E↑X  PI
ST+ X  R↑  /  SQRT  *
RCL 01  /  FC?C 00  RTN
DEG  RCL 00  180  *  SIN
*  PI  X<>Y  /  .END.
```



Übungsteil - Lösung des Problems aus prisma 28,29-81

Sehen wir uns die Lösung von Samuel Martin aus Frankfurt an. Dieser Brief ist einer von denen, die ich gezwungener Maßen übertragen mußte:

Lieber Oliver und Clubmitglieder,

ich finde die Artikel im prisma toll !! Auch ich bin daran interessiert, Programme für Studium, Beruf etc. zu schreiben und dabei einen besseren Programmierstil zu finden. Hier ein Programmvorschlag von mir zur Aufgabe von Wolfgang Leipold aus der Februar Ausgabe:

Prinzip:
Die zu verteilenden "Geldmengen" seien G1 bis G5
für die fünf Personen P1 bis P5
Gesamtmenge: G = G1+G2+G3+G4+G5
Vorhanden sind 7 Einheiten: E1 = 500 DM bis E7 = 1 DM
mit den Häufigkeiten: H1 bis H7
Die Zahl der Scheine (einer Einheit E), die z.B. Person P1 bekommen soll, wird zunächst so berechnet: A1 = INT(G1 x H / G) Nach dieser groben Verteilung der Scheine bleiben einige übrig, die dann einzeln auf die Personen aufgeteilt werden (LBL 04).

Programmablauf:
Das Programm wird gestartet mit LBL^{T}LH. Die Gehälter der Personen werden (am besten in abnehmender Reihenfolge) mit XEQ A eingegeben; danach Einheiten und zugeh. Häufigkeiten (E/H) mit XEQ B. Die Zahl der bisherigen Eingaben wird jeweils angezeigt. Korrektur der Eingaben mit XEQ C: Nach falscher Eingabe (AoderB) XEQ C, dann korrekte Eingabe (AoderB). Mit XEQ D läuft das Programm.

Programmbeschreibung:
LBL 03 verteilt eine bestimmte Einheit auf die einzelnen Personen nach der oben beschriebenen Methode. Wegen der Abrundung (INT) bleiben meistens 2 bis 3 Scheine übrig, die einzeln in LBL 04 verteilt werden. LBL 01 ist die Schleife für die 7 Einheiten. Als Gi werden die verbleibenden Geldmengen genommen. Die Ausgabe der Verteilungen erfolgt i.a. als 7stellige Zahl: abcdefg mit g=Zahl der 500DM Scheine usw. . Falls z.B. g=12 (d.h. g größer als 9) und f=3 sind, dann sieht es so aus: abcde42 statt abcde(3)(12) !!!!! Mit R/S werden nacheinander die Verteilungen P1 bis P5 angezeigt.

Programmdauer:
Das Prgr. läuft unterschiedl. lange. Das Bsp. von Wolfgang hatte ca. 70 sec. gedauert mit den folgenden Verteilungen:

P1: 1 1 0 6 7 8 7 P2: 1 1 0 4 6 6 5 P3: 3 5 4 7 6 5 4
P4: 2 2 3 6 6 6 2 P5: 7 6 3 7 5 5 2

Die Zeilen 53, 54, 85, 124, 125 dienen nur zur Verkürzung der PRGM-Zeit und können auch weggelassen werden. Hier wird abgefragt, ob Gi schon Null erreicht hat und entsprechend weitergesprungen.

Speicherbelegung:
R_{00}, R_{25}: ISG-Speicher R_{01} bis R_{05}: Gehälter von P1 bis P5
R_{06} bis R_{12}: Häufigkeiten der Einheiten R_{13} bis R_{19}: Einheiten (z.B. 500DM bis 1 DM)
R_{20} bis R_{24}: Verteilung der Einheiten auf P1 bis P5

Mit vielen Grüßen Euer Samuel Martin .

```
PRP ..

01•LBL "LH"
CLRG  1  STO 00  STOP

06•LBL A
STO IND 00  VIEW 00  1
ST+ 00  STOP

12•LBL B
STO IND 00  RDN  RCL 00
7  +  X<>Y  STO IND Y
RCL 00  5  -  VIEW X  1
ST+ 00  STOP

27•LBL C
1  ST- 00  STOP

31•LBL D
6.012  STO 25

34•LBL 01
SF 25  5  STO 00
RCL IND 25  RCL 01
RCL 02  +  RCL 03
RCL 04  +  +  RCL 05  +
%

49•LBL 03
ENTER↑  ENTER↑
RCL IND 00  X<=0?
GTO 06  •  INT
ST- IND 25  RCL 25  INT
6  -  10↑X  •  RCL 00
19  +  X<>Y  ST+ IND Y
X<>Y  RDN  RCL 25  INT
6  -  10↑X  /  RCL 25  7
+  RCL IND X  X<>Y  RDN
•  ST- IND 00

65•LBL 06
RDN  DSE 00  GTO 03
XEQ 04  ISG 25  GTO 01
CF 25  "P1:"  ARCL 20
AVIEW  STOP  "P2:"
ARCL 21  AVIEW  STOP
"P3:"  ARCL 22  AVIEW
STOP  "P4:"  ARCL 23
AVIEW  STOP  "P5:"
ARCL 24  AVIEW  STOP

113•LBL 04
20.024  STO 00

116•LBL 05
RCL IND 25  X=0?  RTN
RCL 00  19  -  RCL IND X
X<=0?  GTO 02  RCL 25  7
+  RCL IND X  X<>Y  RDN
-  X<0?  GTO 02
STO IND Y  RCL 25  INT
6  -  10↑X  ST+ IND 00
1  ST- IND 25

144•LBL 02
ISG 00  GTO 05  END
```

Mittwoch, 1.-Mittw...

Hannover Messe '81

CeBIT-Nord, Halle 1

NORD 1 NORD 2 NORD OST NORD WEST OST 1 OST 2 OST 3 SÜD 1 SÜD 2 WEST 1 WEST 2 TAXI BUS

TREFF
Do, 2.4. um 14^{00}
HP-Stand
7705/06

Oliver Rietschel
Hewlett-Packard Anwender-Club
Postscheckkonto Hmb. 48635-207
Postfach 373 · D-2420 Eutin

UTILYS-Teil 2

1. Zusammen mit Ulli Davertzhofen haben wir eine superschnelle <u>Primfaktorenzerlegung</u> ausgearbeitet, die außerdem keinen Komfort vermissen läßt. Sie zerlegt die Fermatsche Zahl F_5 = 4 294 967 297 nach 46 sec. in 641 und weiteren 2 min. 12 sec. in 6 700 417 1 000 003 wird schon nach 1 min. und 10 s als Primzahl erkannt. Ein Programm, das im Großen und Ganzen das gleiche Prinzip verfolgt findet man im HP-67/97 Mathematik-Paket.

<u>2. Zeitschleife.</u> Falls der HP-41C einmal zu schnell arbeiten sollte muß man eine Zeitschleife einbauen. Sehr einfach ist da z.B. das unten angegebene Programm. Es löscht nebenbei auch das ganze Unterprogrammrücksprungregister. Entfernt man eine LBL 00 XEQ 00 Anweisung halbiert sich die Laufzeit. In der unten angegebenen Version braucht das Programm 7.5 s; Mehr als 6 LBL-Anweisungen sind wegen der max. 6 Unterprogrammebenen sinnlos. Setzt man aber statt eben dieser XEQ 00 LBL 00 - Folge mehrere XEQ Befehle vor ein Label, wie im nächsten Vorschlag, läßt sich die Zeitschleife beliebig verlängern !(Im Bsp. arbeitete der HP-41C ca. 1 min !)

<u>3. Hornerschema f, f'.</u> Für Newton und ähnliche Iterationen kann man zur Berechnung von f und f' bei Polynomen das Horner-Schema verwenden, das schnell und rundungsfehlergünstig arbeitet. Die Speicherordnung: In R 00 steht ein Index der Form n+2,002; R 01 enthält nach Programmende f'(x); In R03 ff. werden die Koeffizienten des Polynoms n-ten Grades (n wird für R 00 gebraucht) gespeichert: Das Absolutglied in R02, der Koeff. von x in R03 u.s.w.

```
PRP "PPF"

01•LBL "PPF"
02•LBL A
CF 00  STO 00  FPC  FACT
CLST  STO 01  STO 02
RCL 00  2  MOD  X=0?
XEQ 01  RCL 00  3  MOD
X=0?  XEQ 01  RCL 00  5
MOD  X=0?  XEQ 01
RCL 00

26•LBL 00
RCL 01  7  +  MOD  X=0?
XEQ 01  RCL 00  RCL 01
1  +  MOD  X=0?  XEQ 01
CL 00  RCL 01  13  +
OD  X=0?  XEQ 01
CL 00  RCL 01  17  +
OD  X=0?  XEQ 01
CL 00  RCL 01  19  +
OD  X=0?  XEQ 01
CL 00  RCL 01  23  +
OD  X=0?  XEQ 01
CL 00  RCL 01  29  +
OD  X=0?  XEQ 01
CL 00  RCL 01  31  +
OD  X=0?  XEQ 01  30
+ 01  RCL 01  X↑2
L 00  X>Y?  GTO 00
C 00  GTO 03  1
L 00  X≠Y?  XEQ 04
NDE"  PROMPT

97•LBL 03
"PRIM"  PROMPT

100•LBL 01
FS? 00  GTO 02  RCL 00
LASTX  X=Y?  RTN  SF 00

108•LBL 02
1  ST+ 02  RCL 00  LASTX
/  STO 00  LASTX  MOD
X=0?  GTO 02  LASTX

120•LBL 04
FIX 0  CF 29  " "
ARCL X  CLST  X<> 02  1
X<Y?  "├ ↑"  X<Y?
ARCL Y  FIX 2  SF 29
X<> Z  PROMPT  .END.

START: ZAHL  XEQ A ODER
XEQ PRF. AUSGABE MIT R/S
```

Ralf Pfeifer
Rubensstr. 5
5000 Köln - 50
Tel. 35 20 34

Programm zu 2 →

```
01•LBL A
02 ENTER↑
03 ENTER↑
04 ENTER↑
05 RCL IND 00
06 STO 01
07 DSE 00
08•LBL 00
09 X<>Y
10 ST* 01
11 *
12 RCL IND 00
13 +
14 ST+ 01
15 DSE 00
16 GTO 00
17 *
18 RCL 02
19 +
20 .END.
```

↑ PRGM zu 3

```
01•LBL A
02 XEQ 00
03•LBL 00
04 XEQ 00
05•LBL 00
06 XEQ 00
07•LBL 00
08 XEQ 00
09•LBL 00
10 XEQ 00
11•LBL 00
12 XEQ 00
13•LBL 00
14 .END.
```

```
01•LBL A
02 XEQ 00
03 XEQ 00
04•LBL 00
05 XEQ 00
06 XEQ 00
07•LBL 00
08 XEQ 00
09 XEQ 00
10•LBL 00
11 XEQ 00
12 XEQ 00
13•LBL 00
14 XEQ 00
15 XEQ 00
16•LBL 00
17 XEQ 00
18 XEQ 00
19•LBL 00
20 .END.
```

Ich habe Interesse an folgenden Fachgebieten: Auslegung von Industriekraftwerken, Programm für Teile des I-S-Diagramms (VDI-Wasserdampftafel), Dampfturbinen und Kesselberechnung. Vielleicht gibt es jemanden mit gleichartigen Interessen ???!! E.Barchewitz Wasserwerkstr. 33 68oo Mannheim 31

Wer kann mir die zweite Auflage des Buches:"Richard Eckert,Die Programmierbaren von HP" ersch. im Oldenbourg Verlag 198o für 2 Tage ausleihenoder mir das in diese Auflage neu aufgenommene Kapitel über den 41c gegen Kostenerstattung fotokopieren ? Vielleicht kann ich mich mit Programmen oder anderen Büchern revanschieren. Miloslav Holprecht, Makdorfer Str. 14, 68oo Mannheim 31, Tel. o621/892494

Wollt Ihr ein Teil-Listing der Hex-Code Tabelle ?? Falls ja, so schließt den Drucker an den 41c, Trace Modus und drückt folgende Tasten: Ø STO d 1,o8ooo2 EEX 2o STO d R/S
Happy Programming ! Norbert Weber (44)

Das folgende Programm zeigt die 1o-stellige Mantisse m der in X stehenden Zahl z an, wobei z = s m 1o^{t} mit s = Vorzeicehn von z und o kleiner gleich m kleiner 1o ist und t eine ganze Zahl. Der Stack bleibt erhalten, ebenso das Anzeigeformat. Der Inhalt von L und A (und Q) geht verloren.(z steht in L) Die Mantisse steht in A und in der Anzeige (mit Clear löschen:) LBL "MANT" ABS STO Q CLX RCL d SCI 9 " "
ARCL Q STO d ASTO X ASHF ASTO Q CLA ARCL X ARCL Q CLX LAST X
AVIEW END Johannes Schu (129)

Wir Ihr wißt, war es bisher ein Privileg für Besitzer der Rechner mit der Seriennummer unter 1951, die Befehle wie STO M usw. zu erzeugen. Ich habe nun eine Möglichkeit gefunden, diese Befehle auch auf anderen Rechnern zu erzeugen. Jedenfalls funktioniert es auf meinem Rechner (2o41Ao2868). Und nun zur Realisierung: Zur Erzeugung des Befehls STO M gibt man 1,917585 in das derzeit höchste verfügbare Datenregister, nachdem man vorher MASTER CLEAR (evtl auch CLP) ausgeführt hat. Nun schaltet man in den PRGM Modus um und führt CAT 1 aus, welches man bei dem ersten Erscheinen von .END. unterbricht. Dann muß DEL größer als Null ausgeführt werden. Als letztes wird XEQ END ausgeführt. Der erzeugte Befehl befindet sich nun im Programmspeicher. Sollte dies nicht der Fall sein, so ist die o.g. Zahl außerdem in R oo zu speichern. Andere Befehle lassen sich analog erzeugen (siehe prisma 3/5-8o) Martin Klingenberg (232)

Böse Worte !

Je größer ein Club wird, desto größer wird die Zahl derjenigen, die ihre Freude an übler Nachrede finden. Um solchen Subjekten zumindest die "Arbeit" zu erschweren hier ein paar Worte der Richtigstellung:

Sämtliche Behauptungen, der Club oder einzelne Mitglieder würden die Berichte in prisma nur irgendwo abschreiben oder übersetzen sind frei erfunden und entbehren jeder Grundlage. Sollte etwas einmal doch übernommen werden (z.B. das KA-Programm vom PPC), was mit Sicherheit extrem selten ist, so wird dieses ausdrücklich erwähnt.

Erlogen ist die Behauptung, Matthias Grabiak hätte sein Wissen aus dem PPC-Journal entnommen. Richtig ist vielmehr, daß Matthias zum Zeitpunkt, als er den Bericht schrieb, noch nichts von der Existenz des PPC wußte. Darüber hinaus brachte er ja auch neues, was bis heute nicht im PPC-Journal erschienen ist. Das gilt nicht nur für Matthias !!

Eine böswillige Unterstellung clubfeindlicher Elemente ist, daß finanzielle Mittel des Clubs zu clubfremden Zwecken verwandt werden. Die Verwendungszwecke sind im Impressum genau aufgeführt. Es wird für jedermann einsehbar Buchführung geführt. Sofern überhaupt Aufwandsentschädigungen für die, das sei betont, stupiden und anödenden organisatorischen Arbeiten gezahlt werden, so in der Regel nicht mehr als 6,--DM je Stunde. Das ist extreme Unterbezahlung, wenn ich mit den 22,5o DM und mehr vergleiche, die ich für Nachhilfestunden bekomme !

Wer solche Behauptungen weiterhin verbreitet, muß damit rechnen, daß ich auf der nächsten Mitgliederversammlung seinen Ausschluß aus dem Verein beantrage !

iloslav Folprecht, Maxdorfer Str. 14, 6800 Mannheim 81, T.0621/892494
lub Nr. 345

ieber Oliver! Liebe Mitglieder!

eim Erhalt der Nachdrucke des Prisma 1980 habe ich zum erstenmal
n Zusammenhang mit dem Club etwas bedauert - daß ich nicht schon
iel früher beigetreten bin !
eute habe ich auch einen Brief vom Frank Altensen bekommen, dem
ch seinerzeit ein Buch (H. Alt : Angewandte Mathematik, Finanz-
athematik, Statistik, Informatik für UPN-Rechner) für die Club-
ibliothek angeboten habe. Er fragt nach dem Preis. Da ich es
oppelt habe kostet das Buch den Club natürlich nichts. Ganz selbst-
os bin ich dabei aber nicht : ich hoffe, daß ich und alle Mitglie-
er den Nutzen einer gut ausgebauten und funktionierenden Club-
ibliothek genießen werden.

BATTERIEN UND STROMVERBRAUCH DES HP 41 C

ngeregt durch Oliver's Artikel im Prisma 10/80 S.7 habe ich den
tromverbrauch meines Rechners gemessen :

	I_{OFF}	I_{ON}	I_{ON} m. Pr.	I_{ON} m.Pr.+BEEP
l C ohne Peripherie	0.001	0.5	5.3	5.5
l C + 1 Memory	0.0015	0.503	6.4	6.6
l C + 3 Memorys	0.002	0.504	8.2	9.0
l C + 3 M + KL	0.0057	0.51	9.3	9.5
l C + 3 M + KL + DR	0.012	0.52	9.6	9.8

lle Ströme in mA bei 6 V am Netzteilanschluß, gemessen mit digi-
alem Meßgerät der Genauigkeitsklasse 0.1 %. Bei 4.6 V am Netz-
eilanschluß erscheint BAT Anzeige, bei Verwendung des Kartenlesers
erweigert dieser schon bei weniger als 4.8 V den Dienst. Bei Be-
eb mit eingelegten Batterien oder Akkus sind die Verhältnise güns-
ger : BAT erscheint bei 4.25 V und der Kartenleser arbeitet noch
i 4.3 V einwandfrei. Dies ist dadurch bedingt, daß Netzteilbuchse
d Batterieanschlußkontakte keine direkte Verbindung haben, sondern
r + Pol der Netzteilbuchse über ein Schutzelement, an dem je nach
romstärke 0.3 bis 0.5 V abfallen, mit der Rechnerschaltung verbun-
n ist.
ch eine interessante Eigenschaft, die von sehr guter Funktion
r Rechnerstromversorgung und interner Stabilisierung zeugt :
i kleineren Spannung steigt der Strom, die aufgenommene Leistung
max. ca. 60 mW) bleibt bis zur Abschaltgrenze gleich.

Der Kartenleser verbraucht beim Karteneinlesen oder -Beschreiben ca. 23o mA, beim abgeschalteten Rechner erhöht sich der Ruhestrom nur um ca. 3 uA, wenn man nicht vergessen hat den Rechner mit aufgesetztem KL ein- und wieder auszuschalten. Unterläßt man dies, so liegt der Ruhestromverbrauch bei ca. 120 uA und daß kann schon die Batterielebensdauer beeinträchtigen.
Eine Mallory Alkali-Mangan-Lady-Batterie hat eine Kapazität von 580 mAh, ausreichend, um im RUN-Betrieb mit voller Peripherie 58 Stunden lang zu rechnen. Bei einem I_{OFF} = 0.012 mA würde sie theoretisch ganze 48000 Stunden, das sind 2013 Tage, halten, wenn wir nicht zwischendurch auch rechnen werden. So hat der Ruhestrom kaum einen Einfluß auf die Batterielebensdauer, wenn er sich im Bereich um die 10 oder 20 uA hält. Wir können auch unbesorgt den Kartenleser immer am Rechner aufgesteckt lassen, die 3 eingesparte uA würden uns praktisch gar nichts bringen.
Wesentlich wichtiger ist hier die Frage, ob die neu gekauften Batterien auch wirklich ihre 580 mAh Kapazität haben oder schon so lange gelagert sind, daß uns nur ein Bruchteil davon zur Verfügung steht. Bei öfteren Verwendung des KL kann man dann die Batterienlebensdauer nicht in Monaten, sondern in Wochen zählen.
Überraschend war für mich, daß der Tongenerator nur ganze 0.2 mA verbraucht und das nur während der Tonerzeugung-kaum nennenswert für den Gesamtstromverbrauch im RUN-Betrieb.

WIEDERAUFLADBARE ZELLEN

Bei diesen ist fast unbekannt (siehe Prisma 81, S. 48 unten), daß es zwei für uns in Frage kommenden unterschiedlichen Typen gibt : NiCd-Zellen und Masse-Zellen. NiCd-Zellen gibt es schon relativ preiswert (um ca. 5 DM), sie haben aber den Nachteil der ziemlich hohen Selbstentladung. So hat eine NiCd-Zelle bei 20°C nach 2 Wochen nur noch 75% der Anfangskapazität, nach 1 Monat 60%, nach 2 Monaten 35% und nach 3 Monaten nur noch 20%. Bei 30°C hat sie (und der Sommer kommt auch) nach 1 Monat nur noch 40%, nach 9 Wochen gar nichts mehr von der Anfangskapazität behalten.
Die Masse-Zellen haben eine sehr geringe Selbstentladung, nach 10 Monaten Lagerung bei Raumtemperatur sind noch 50% der Kapazität vorhanden. Von der Fa. VARTA gab es diese Zellen bis vor kurzem unter der Bezeichnung 151D zu einem Endpreis von ca. 12 DM/Stck. Wegen

zu teurer Herstellung wird von der Fa. VARTA nur noch die NiCd Version 151 RS hergestellt. Der Preis bewegt sich für diese NiCd Zellen bei ca. 7-8 DM, allerdings sind sie mit den oben aufgeführten Nachteilen der NiCd-Zellen behaftet.
Beide Typen, 151D und 151RS, haben eine Nennkapazität von 150 mAh. Man muß sie also theoretisch 4x so oft wechseln wie normale Batterien. Da wir sie aber, im Gegensatz zu normalen Batterien, immer voll aufladen können, fällt in der Regel die Relation viel günstiger aus.
Nun, das klingt alles ganz schön, aber wie kommen wir an die Masse-Zellen ran, werden sie sicher fragen. Ich habe, nachdem ich mich mit Netzteil und zwei Stolperkabeln oder einem Batteriegehäuse für 4 Monozellen und einem Stolperkabel abgeplagt habe, 8 von diesen Massezellen und ein Steckerlader angeschafft und bin für die nächsten Jahre alle Batteriesorgen los. (Zugegeben : den letzten nicht ganz verbrauchten Batteriesatz habe ich aufgehoben - man weiß nie wann man das Aufladen vergißt. Passiert ist es mir aber nur einmal!). Und das ganze für 68 DM!
Zu diesem Preis + 5 DM Versand- und Verpackungkosten kann ich noch einige Sets, bestehend aus 8 Stck. orig. VARTA Massezellen 151D und einem VARTA Steckerlader für 151D, besorgen. Und damit ich keine Arbeit habe, richtet bitte Eure Bestellungen an meine Frau :
Ursula Folprecht, Maxdorfer Str. 14, 6800 Mannheim 81, T. 0621/892494.
Und wer schon einen Satz hat, hier noch die Einzelpreise :
151D 6 DM, Steckerlader VARTA 25 DM. Versandtwird (Vertrauen in Clubmitglieder) gegen Rechnung.

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Danke für den Bericht von Norbert Weber über DDM. Habe zuerst ein DDM ohne die geringsten Schwierigkeiten gebaut. Kann man auch jetzt gekaufte Memorys so umbauen? JA ! Beachtet aber den Hannovermesse-Bericht !

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Habe auch einige Bücher über Progr. Rechner oder mit Programmen. Muß aber zuerst ein Verzeichnis anfertigen. Es wäre gut, Oliver, wenn Du in etwa die Abschlußtermine für die nächsten Prisma-Ausgaben angeben könntest. Natürlich: Jeweils etwa 4 Wochen vor dem Erscheinungsmonat (+-2 Wochen). Unverbindlich ! Termine siehe prisma 44-81 oben ! Nächste Ausg.: August, September, Oktob.

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Hannover 1981 - Die Messe der Messen

Am Do,d.2.4.81 war es soweit: Einige Minuten vor 14^00 Uhr entstand ein Menschenauflauf in unmittelbarer Nähe des HP-Taschenrechner-Standes. Beim Durchzählen wurden etwa 35 HP-41-Menschen registriert, die offenbar der Einladung im letzten prisma gefolgt waren. Viele hatten auch meinen Vorschlag einer Namensschild-Magnetkarte aufgegriffen und Walter Pieperhoff hatte gleich zwei Hände voll Pappmagnetkartenansrecknadeln gefertigt, die reißenden Absatz fanden.
Pünktlich um 14^00 Uhr folgten wir der Einladung von HP in die über der Ausstellungsebene liegenden Besprechungsräume, wo nach einer Bekanntmachung mit mehreren HP-Repräsentanten unsere Aufmerksamkeit einem ausführlichen Bericht über die derzeitige Situation und die zu erwartenden Neuentwicklungen auf dem uns interessierenden Sektor galt. Auf der Messe selbst wurde offiziell noch keine weitere Peripherieeinheit vorgestellt, wenngleich der Hinweis auf die ständig fortlaufenden Entwicklungsarbeiten nicht fehlte.

Der weitaus größte Teil der bis zu mehr als 3 Stunden dauernden Unterredungen setzte sich aus persönlichen Fragen und Gesprächen zusammen, die größtenteils zur Zufriedenheit der Mitglieder ausfielen. Aus diesem Grunde ist es aber auch nicht möglich, hier über alle Einzelheiten zu berichten. Durch die vielen Gespräche und Diskussionen wurde ein wesentlich besseres Verhältnis zu HP geschaffen, wobei zu betonen ist, daß dies vornehmlich durch das freundliche Entgegenkommen von HP bedingt ist und nicht durch Rückstellung unserer Forderungen.
Unmittelbarer Nutzen für Clubmitglieder ist in dieser Ausgabe z.B. die Erlaubnis von HP, die Kurzanleitung des Magnetkartenlesers und den im deutschen Handbuch fehlenden Anhang der Dokumentation zum BarCode-Leser abzudrucken. Viele andere Vorteile, die - das sei hier ausdrücklich betont - bei einem Konfrontationskurs gegen HP niemals möglich gewesen wären, sollen hier nicht besprochen werden, sondern werden sich in unmittelbarer und näherer Zukunft positiv für alle Mitglieder auswirken.
Ein weiterer Beschluß wirkt sich bereits in diesem Bericht aus: In unserem prisma sollen ab sofort vornehmlich Dinge stehen, die bereits Hand und Fuß haben: Spekulationen und nicht feststehende Dinge werden zukünftig vornehmlich unter den aktiven Mitgliedern besprochen, die diesen Club tragen und ohnehin jetzt schon in aktiver Kommunikation untereinander stehen. Jedes andere Mitglied kann auch gerne mitarbeiten; um nur ein paar Beispiele zu nennen: Berliner Ortsgruppe, wer leitet sie ? Statik-Programme, wer bringt Ordnung hinein ?
Weiterhin wurden Ergänzungen zu meinem neuen Satzungsentwurf besprochen und über zukünftige Projekte diskutiert. Abschließende Besprechungen und eine echte Mitgliederversammlung waren jedoch aufgrund des allgegenwärtigen Messelärms nicht möglich. Die nächste Mitgliederversammlung ist für Juli vorgesehen; die Adressen der Teilnahmewilligen liegen mir ja vor; der Ort und Zeitpunkt steht aber noch nicht genau fest.
Abschließend hat Walter Pieperhoff noch ein wenig Atmosphäre einzufangen gesucht und u.a. die Bilder der nächsten Seite geschossen. Weil ein Breitwinkelobj. leider fehlte, mußte er damit warten, bis sich die Reihe der HP-41-Fans ein wenig lichtete, so daß nur ein paar der wirklich begeisterten Fans sich wiederfinden werden. Dennoch: Auch das längere Ausharren brachte Vorteile; sei es nur, daß man ein paar Worte mit dem "First Man" von HP, der zu späterer Stunde direkt aus den USA einflog, wechselte .....

Neue DDM- Technik !!!

Vor einem Jahr veröffentlichten wir einen Memory-Modul Umbau von Norbert Weber . Detlev Bock (2) und Nicht-Mitglied Uwe Zierfuß bieten jetzt einen verbesserten Umbau an: Zwei Speichermodule können zu einem DDM umgebaut werden, ohne daß zusätzlich ein drittes Modul^x vorhanden sein muß, wie es bisher der Fall war. x=Speichermodul
Beispiel: DDM in Port 1 / ROM in Port 2 / Drucker Port 3 / Kartenleser Port 4
Bei zwei DDMs adressiert der 41c insges. 319 Register und läßt die Ports 3 und 4 für weitere Peripherie frei. Es werden zwei Leistungen angeboten:
1) DDM-Umbau total: Gegen Zusendung von 2 Memorys 1 DDM zurück. Preis: 4o,--
2) Umbau DDM Weber oder Lütke Uphues: Gegen Zusendung des alten Umbaus ein neuer zurück: Preis: 2o,-- // Bei 1) wird volle Garantie übernommen, bei 2) nur auf den Umbau.
Umbau/Vertrieb innerhalb einer Woche. Die Module bitte in einer Schachtel an Detlev Bock(2) schicken, Geld beilegen oder überweisen: Commerzbank Göttingen, Kto. 63o 64o 1



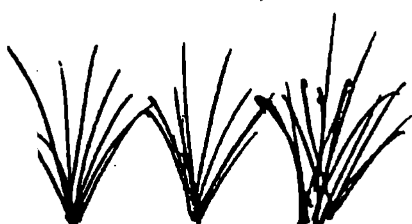
Ein Gespräch sagt mehr
als ein dutzend Briefe:

Klaus Werner Hoenow (11)
Ulrich Merz (63)
Walter Bierwirth (1o5)

(von rechts nach links)

Wenngleich nicht alle 35 Mitglieder einen Platz fanden: Freundliche HP-Bar-Damen sorgten dafür, daß die Kehle nicht zu trocken wurde....und das Stehen erträglich.

Oliver Rietschel (1)(2.von rechts) im Gespräch mit dem Ehegatten von Annegret Jäger (312)(1.von rechts)
Die wenigen Stunden vergingen wie im Flug: man hätte sich noch Tage unterhalten können!!!







Stunden nach den Reden der HP-Genies beginnt sich die Reihe der 41er Fans zu lichten: Die Züge fahren ab.
Im Vordergrund von links nach rechts:
Detlev Bock (2)
Walter Bierwirth (1o5)
Ulrich Merz (63)
Klaus Werner Hoenow (11)

Leider fehlte uns ein Breitwinkelobjektiv für eine richtig monumentale Aufnahme aller 35 ....

Dieser Brief ist die Antwort auf die von Sven gestellte Aufgabe einen 8-Stellen D-H-Wandler zu bauen(Prisma 100-81). Schwierigkeiten gab es nur mit der LN Fkt. die im Bereich 16^8-12 bis 16^8 den gleichen Wert liefert, so daß die Schleife einmal mehr durchlaufen wird. Dies korrigiert PRGM-Schritt 37. Der H-D-Wandler von Sven funktioniert bei mir nicht. Grund: Nach Ausführung von ASTO Befehlen entspricht dem 2. Digit des entsprechenden Reg. das 2. Digit des 7. Bytes im A'Reg.(Weinachtsheft Matthias) Es ist also noch CF 06 einzufügen. Außerdem habe ich y^x aus der Schleife genommen. Beide PRGMe brauchen jetzt für 8 St.(Hex) ca. 10 sec.

```
01◆LBL "D-H
02 CLA
03 ENTER↑
04 ENTER↑
05 LN
06 16
07 STO Z
08 LN
09 /
10 INT
11 CHS
12 STO [
13 CHS
14 Y↑X
15◆LBL 00
16 STO T
17 MOD
18 STO T
19 -
20 RCL Y
21 /
22 9
23 -
24 X>0?
25 ISG X
26 ""
27 139
28 +
29 0
30 X<> \
31 X<>Y
32 X<> [
33 "⊢*"
34 X<>Y
35 STO \
36 CLX
37 STO ]
38 +
39 "⊢*"
40 STO [
41 RDN
42 16
43 /
44 ISG [
45 GTO 00
46 RCL ]
47 RCL \
48 CLA
49 STO [
50 RDN
51 STO \
52 AVIEW
53 END
```

```
01◆LBL "H-D
02 16
03 1/X
04 0
05◆LBL 10
06 RCL d
07 "⊢◆◆◆◆x̄x̄
08 0
09 X<> \
10 ASTO d
11 CF 03
12 CF 06
13 STO [
14 CLX
15 X<>Y
16 X<> d
17 +
18 X=0?
19 GTO 09
20 40
21 -
22 X>0?
23 DSE X
24 ""
25 10
26 +
27 16
28 ST* T
29 CLX
30 RCL T
31 *
32 +
33 GTO 10
34◆LBL 09
35 RDN
36 END
          CAT 1
LBL'D-H
END
86 BYTES
LBL'H-D
END
67 BYTES
```

Viel Spaß bei der weiteren Verbesserung dieser PRGMe wünscht:

Jürgen

Zum letzten Mal: SIZE-ROUTINE Type: Schnellstmöglich !

Lieber Oliver ! Liebe Clubmitglieder !

...... Ich habe mir einmal das Register c näher angeschaut und als Abfallprodukt entstand ein Programm, das den SIZE und den verbrauchten Programmspeicherplatz in Registern direkt berechnet, wodurch es natürlich sehr schnell ist: Nach ca. 2 Sekunden zeigt es den SIZE an, nach ca. 3,5 den Verbrauchten Platz. Der einzige Nachteil ist, daß es leider so viel Platz braucht, um so schnell zu sein.

Zu Beginn muß jeweils die Anzahl der Memory-Module eingegeben werden (0,1,2,3,4); schneller wird es, wenn man die Zeilen 13 bis einschl. 18 wegläßt und anstatt X()N in Zeile 51 STO N gefolgt von der Zahl 256 für kein Modul, 32o für 1 M., 384 für 2 M., 448für 3 M., 512 für 4 Module (oder 41cv).

Diese Eingabe hat folgenden Hintergrund: Der Rechner verzeichnet in drei Digits in Register c (Reg ØØ) wieviel Register für Daten zur Verfügung stehen und zwar zählt er rückwärts, beginnend jeweils mit der entsprechenden oben angegebenen Zahl. (Vergleiche auch Rechnerorganisation Teil 4) Bei max SIZE hat er demzufolge, egal bei welcher Konfiguration, bis 193 heruntergezählt; der SIZE ergibt sich also aus der Differenz der beiden Zahlen. Das .END. ist auf die gleiche Weise verzeichnet; bei REG ergibt die Differenz, subtrahiert vom jeweiligen SIZE, das erste Summenregister. Beispiel: 2 Memorys ergeben 384,SIZE oo8, REG 11, verzeichnete Zahl ist $183_{16}=387_{10}$ und 8-(384-387)=8-(-3)=11

Da das .END. am Ende des letzten Programms steht, ergibt sich der verbrauchte Platz als folgende Differenz: (RegØØ)-(.END.) Da jedes angefangene Register gezählt wird, hat man meist ein paar Byte mehr frei. Zusätzlich tritt eine weitere Schwankung dadurch auf, daß das .END. verschieden viele Bytes verbrauchen kann.

Andreas Marktscheffel (61)

```
LBL'S
END       159 BYTES
01+LBL "S"
02 2
03 "AA"
04 RCL [
05 STO \
06 RCL c
07 STO [
08 ASHF
09 "├AAAA"
10 RCL d
11 X<> [
12 STO d
13 R↑
14 4
15 +
16 64
17 *
18 STO \
19 CLST
20 FS? 02
21 512
22 FS? 03
23 256
24 FS? 04
25 128
26 +
27 +
28 +
29 FS? 05
30 64
31 FS? 06
32 32
33 +
34 +
35 FS? 07
36 16
37 FS? 08
38 8
39 +
40 +
41 FS? 09
42 4
43 FS? 10
44 2
45 FS? 11
46 1
47 +
48 +
49 +
50 ENTER↑
51 X<> \
52 X<>Y
53 -        VIEWX
54 STO ]
55 CLST
56 FS? 14
57 512
58 FS? 15
59 256
60 FS? 16
61 128
62 +
63 +
64 +
65 FS? 17
66 64
67 FS? 18
68 32
69 +
70 +
71 FS? 19
72 16
73 FS? 20
74 8
75 +
76 +
77 FS? 21
78 4
79 FS? 22
80 2
81 FS? 23
82 1
83 +
84 +
85 +
86 RCL [
87 STO d
88 RDN
89 RCL \
90 X<>Y
91 -
92 RCL ]
93 END
```

BARCODELESER - DEMONSTRATIONSPROGRAMME

Unser Dank für die Erlaubnis, die Demonstrationsprogramme für den optischen Lesestift, die im deutschen Handbuch fehlen, zu übersetzen und abzudrucken, gilt unseren Gesprächspartnern von HP auf der Hannovermesse. Hier also die - verkürzte - Übersetzung:

S E A R C H   A N D   D E S T R O Y Suche und zerstöre

Du bist der Kapitän eines U-Boot-Jägers und sollst feindlichen U-Booten die Hölle heiß machen ! In Deinem Operationsgebiet von der Größe 10 x 10 Flächeneinheiten befindet sich z.Zt. genau ein U-Boot. Um die Position Deines Zerstörers zu definieren lese eines der als Barcodes dargestellten Felder (1Feld=1Flächeneinheit) mit dem Lesestift ein. Von dieser Ausgangsposition aus suche das feindliche U-Boot mithilfe Deines Echolotes und den damit in Zusammenhang stehenden Meldungen Deines "Navigationsrechners /':" sowie den Piepstönen des Echolots. Je höher der zweite Ton - er ist nur zu vernehmen, wenn das U-Boot in Reichweite ist -, desto näher ist das U-Boot. Versuche, mithilfe der BarCodes direkt über das U-Boot zu gelangen. Dann werfe Deine Wasserbomben ab (einlesen des Barcodes DROP DEPTH CHARGES) ! Bist Du genau über dem Boot, wirst Du es versenken; ist es innerhalb einer Reichweite von 1,5 Feldern, so besteht die Möglichkeit, daß es sinkt.
Nach dem Abwerfen der Bomben gibt es einen großen BUMS und der Rechner zeigt, wenn auch das U-Boot bums gemacht hat, BOOM an, ansonsten MISSED und die Jagd geht weiter. Nach dem Werfen einer Wasserbombe bewegt sich das U-Boot um ein Feld in eine beliebige Richtung.

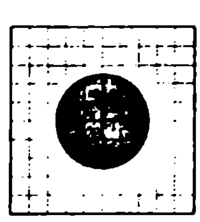
Spielen:
1) Lese die BarCodes unter Search and Destroy ein und danach "XEQ SG"
2) eine beliebige, ganzzahlige Zufallszahl eingeben.
3) Die Anzeige enthält die Position des Schiffes. Mithilfe des BarCode lesers kann man zu jedem beliebigen anderen Feld fahren. Ein Echo des ausgesandten Sonartones erklingt, wenn das U-Boot 2,5 oder weniger Felder vom Zerstörer entfernt ist.
4) Mit "STATUS" wird angezeigt, wieviel Bewegungen und Wasserbomben Du bisher benötigt hast.

Example:

Operation	Display	
Scan [XEQ] SG	*SEED?*	
Key in the integer seed 73, then scan BEGIN	*ORDERS 0*	
First move:		
Scan 55	*SCANNING*	(no echo)
	ORDERS 55	

Since there was no echo, eliminate the shaded region below.

Second move:		
Scan 22	*SCANNING*	(low pitch echo)
	CONTACT	
	ORDERS 22	



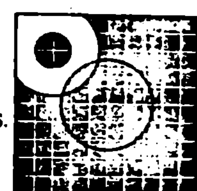
Since a low pitch echo sounded, eliminate the area very close to the ship's location.

Third move:

Scan 20 *SCANNING* **(high pitch echo)**

CONTACT

ORDERS 20



Since a high pitch echo sounded, the sub is close. Scan DROP DEPTH CHARGES.

BOOM

1 CAN

3 SCANS

GOOD JOB

NEW GAME

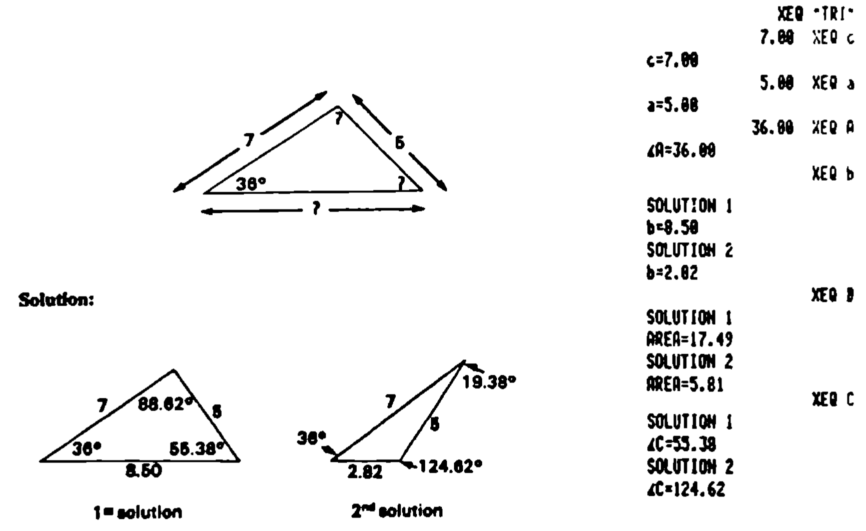
ORDERS 0

Dreiecksberechnungen T R I A N G L E S O L U T I O N S

Es gibt - glaube ich - nicht viel zu den ganzen Dreiecksprogrammen zu sagen. Doch hier zur Demonstration ein Prgr. bei dem kein Tastendruck erforderlich ist: Sind von einem Dreieck 3 Seiten und/oder Winkel bekannt, so können die fehlenden Seiten und Winkel, sowie die Fläche des Dreiecks berechnet werden.
Die Daten werden über das Strichcodetastenfeld eingegeben und danach das in der Grafik zu diesem Wert gehörige Strichcodefeld eingelesen. Nach drei Eingaben lassen sich die restlichen Werte berechnen.

Anleitung:
1) Lese die Strichcodes TRIANGLE SOLUTIONS ein ! Dann lese ein: "XEQ TRI"
2) Lese den ersten Wert auf dem Barcodefeld ein. Winkel in Altgrad. Dann den Dreieckstell.
3) Wiederhole Schritt 2) dreimal.
4) Lese den Strichcode eines beliebigen Dreieckstells ein und du erhältst die Lösung ! Weiter bei Schritt 2) oder 4)

Example: Solve the triangle below.



P A C E Wie schnell soll ich Barcodes einlesen ?

Nehme Deinen Griffel zur Hand und führe das Programm (LBL o1 PSE TONE 7 GTO o1) aus. Wenn Du jetzt mit Deinem Griffel - ohne diesen einzuschalten - über irgendwelche Barcodes fährst, sollte Dein Rechner genau nach jeder Zeile einmal piepen. Dann ist die Geschwindigkeit in etwa optimal und die Lesefehler am seltensten.

Das Programm Character Builder wurde nicht übersetzt und abgedruckt.



"SYNTHETIC PROGRAMMING ON THE HP-41 C"
W.C. Wickes

Synthetische Programmierung, für den einen ein Buch mit sieben Siegeln, für den anderen Spielerei und für den nächsten D I E Programmierungsmethode zur Optimierung von Programmen, ist in unserem Club bisher immer nur "Artikelweise" behandelt worden. Das heist für den Anwender: wiederholtes Nachschlagen nach den entsprechenden Artikeln.
Der Autor des Buches "Synthetic Programming on the HP-41 C", W.C. Wickes, hat dem nun quasi ein Ende bereitet. Das Buch umfaßt sowohl die Einführung als auch die Anwendung der Syntetischen Programmierung und ist damit sowohl für den "Einsteiger" als auch für den Fortgeschrittenen besonders zu empfehlen. Für erstere stellt es ein didaktisch gut aufgebautes Lehrbuch dar, wobei man nie den Eindruck hat, daß der Autor mit erhobenem Zeigefinger auf einen blickt, und für zweitere ist es wohl eher als Katalog oder Nachschlagewerk zu empfehlen. Einen winzig kleinen Nachteil hat aber auch dieses Buch: es ist in Englisch geschrieben! Dieser "Nachteil" läßt sich - meiner Erfahrung nach - mit ein paar grundlegenden Englischkenntnissen, einem Englisch-Deutsch Wörterbuch und einigen Erfahrungen mit dem 41er recht schnell beheben.
Das Buch(man sollte eher "Heft" dazu sagen, da es DIN A4 Format hat und den HP-Solution-Books ähnelt) ist in sieben Kapitel unterteilt. Es hat drei Anhänge, zehn Figuren/Bilder und sechs Tabellen.
Auf vielfachen Wunsch, seien hier die sieben Kapitel mit den Hauptüberschriften angegeben:

1. Why's and Wherefore's(Warum und Wofür)
2. Inside the HP-41 C(Innerhalb des Rechners)
3. Exotic editing with the Byte Jumper(Außergewöhnliche Korrekturen mit dem "Byte Jumper")
4. The Status Register
5. Programs for programming(Programme zum programmieren)
6. Applications(Anwendungen)
7. Amusing Anomalies(lustige Unregelmäßigkeiten)

Kapitel 1 und 2 beinhalten eine philosophische und basistechnische Einführung in das Thema der synthetischen Programmierung, d.h. es wird erklärt, warum überhaupt synthetische Programmierung und worauf diese(von der Hard- und Software) aufbaut. Derjenige, der mehr Softwareintressiert ist und auf die Hardware oder auf die Erklärung warum etwas funktioniert verzichten möchte, kann diese Kapitel ruhig überschlagen. Jeder Schritt/Befehl der synthetischen Programmierung wird in den nachfolgenden Kapiteln themenspezifisch genau erklärt.

Die nachfolgenden Kapitel sollte man allerdings der Reihe nach durcharbeiten, da jedes weitere Kapitel auf Befehle/ Methoden/Programmsequenzen von anderen Kapiteln aufbaut. Intressant ist auch, daß nicht nur gezeigt wird, daß es diesen oder jenen Befehl gibt und wie er funktioniert, sondern daß die Befehle in vollständig dokumentierten Programmen verwendet werden und dort zeigen, was sie "können". Das Spektrum der Programme reicht von der synthetischen Erzeugung von bel. Alpha-Zeichen, über die Umwandlung von Alpha-Zeichen in numerische Werte und umgekehrt, über ein Key-Assignment-Programm, bis hin zu einem Hang-Man-Spielprogramm(entspricht dem sog. "Galgenspiel"). Ein ganz besonderer Clou für Spiele-freaks dürfte die beschriebene Möglichkeit der Erzeugung von 128(bzw.119) verschiedenen Tönen sein, wodurch sich Geräusche erzeugen lassen, die das Herz jedes "Flipper" oder "Space War" Spielers höher schlagen lassen(Bei mir hat das soweit geführt, daß mein Rechner neben dem Metronom steht und gleichmäßig "mittickt") - und das alles mit jedem beliebigen HP-41 C(nicht an Bug2 Maschinen gebunden).
Das Konzept des Buches ist aber so, daß primär die praktische Anwendung der synthetischen Programmierung erläutert wird, wie etwa Tips, um a l p h a n u m e r i s c h zu sortieren etc.

Im Anhang des Buches finden sich dann alle längeren Prgm.s, die in dem Heft enthalten sind in Form von Barcodes als auch eine Barcode-Zeichen-Tabelle(alle Alphazeichen als Barcodes). Außerdem wird dem Leser dort die Grundlagen von Zahlensystemen erläutert.

Ein weiteres Plus des Buches ist, daß es die Anwendung der synthetischen Programmierung auch auf die Peripheriegeräte bezieht, wie etwa das "Springen" zu einer bel. Stelle innerhalb eines ROM's oder die Programmierbarmachung von nicht-programmierbaren Befehlen. Allerdings werden nicht die Möglichkeiten aufgezeigt, die der Bar-Code-Leser in Bezug auf die synthetische Programmierung eröffnet, was wahrscheinlich darauf zurückzuführen ist, daß das Gerät erst in letzter Zeit auf dem Markt erschienen ist.

Als abschließende Bewertung kann man das Buch allen empfehlen, die sich mit der synthetischen Programmierung beschäftigen oder beschäftigen wollen. Es stellt sowohl eine optimale Einführung als auch ein sehr umfangreiches Nachschlagewerk dar.

Das Buch ist zu erhalten von:
Larken Publications
P.O. Box 987
College Park MD 20740 U.S.A.

Preis: 11Dollar 50Cents

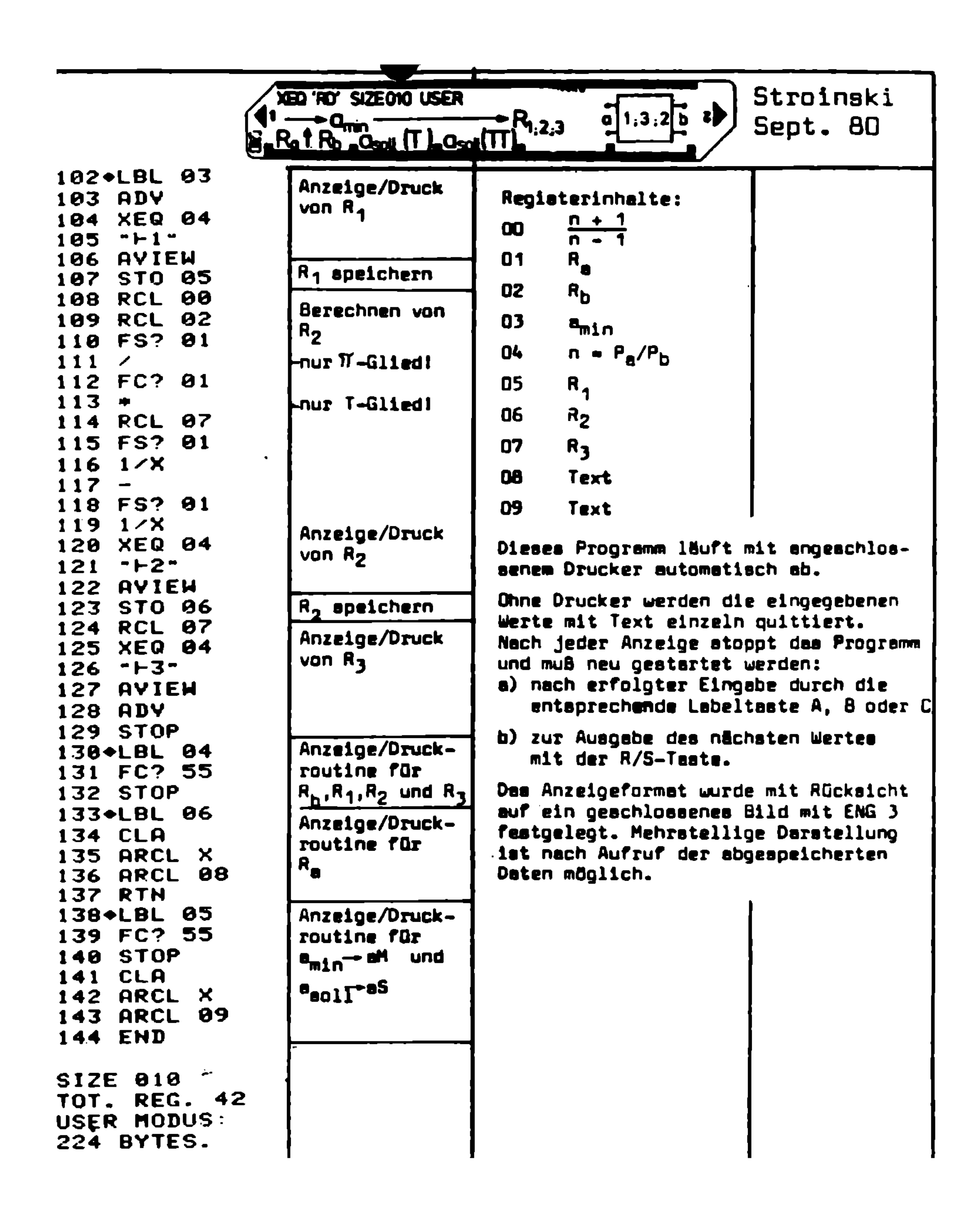
Das Buch kann auch für acht bis 14 Tage von der Clubeigenen Bibliothek ausgeliehen werden!!
Frankierter(1,40 DM) und selbstaddressierter Rückumschlag(Versandtasche, größer als DIN A4) an:

Frank Altensen
Uhlandstr. 9
6365 Rosbach III

Viel Spaß bei der Lektüre und Happy Programming

Ohm'sche Dämpfungsglieder (weiterentwickelt für HP-41C aus EE1-11 für HP67)

Programm	Kommentar
01◆LBL "RD" 02 "OHM=R" 03 ASTO 08 04 " dB=a" 05 ASTO 09 06 SF 27 07 ENG 3 08 STOP	Haupt-Label, Vorbereitungsschritte
09◆LBL A 10 STO 02 11 X<>Y 12 STO 01 13 ADV 14 XEQ 06 15 "⊢a" 16 AVIEW 17 X<>Y 18 XEQ 04 19 "⊢b" 20 AVIEW	Speichern, Anzeige/Druck der Eingabewerte R_a und R_b
21 X>Y? 22 X<>Y 23 / 24 ENTER↑ 25 ENTER↑ 26 1 27 - 28 SQRT 29 X<>Y 30 SQRT 31 + 32 LOG 33 20 34 * 35 STO 03	Berechnen des durch R_a und R_b gegebenen Mindest-Dämpfungsmaßes a_{min} a_{min} speichern
36 XEQ 05 37 "⊢M" 38 AVIEW 39 ADV 40 STOP	Anzeige/Druck von a_{min}→aM
41◆LBL B 42 "T" 43 AVIEW 44 CF 01 45 XEQ 02 46 * 47 - 48 CHS 49 GTO 03	Schaltung: T-Glied, Anzeige/Druck, Rahmenprogramm
50◆LBL 02 51 XEQ 05 52 "⊢S" 53 AVIEW	Anzeige/Druck von a_{soll}→aS
54 RCL 03 55 X>Y? 56 GTO 09	Prüfung: aS > aM
57 X<>Y 58 10 59 / 60 10↑X 61 STO 04	Berechnen des Leistungsverhältnisses $n = P_a/P_b$ n speichern
62 FS? 01 63 1/X 64 RCL 01 65 * 66 RCL 02 67 * 68 SQRT	Berechnen des Koppelwiderstandes R_3
69 FC? 01 70 2 71 FC? 01 72 *	nur für T-Glied
73 RCL 04 74 1 75 -	
76 FS? 01 77 * 78 FS? 01 79 2	nur für π-Glied
80 / 81 STO 07	R_3 speichern
82 1 83 RCL 04 84 + 85 LASTX 86 1 87 - 88 / 89 STO 00	Berechnung von $\frac{n+1}{n-1}$ speichern
90 RCL 01 91 RTN	Rücksprung ins Rahmenprogramm
92◆LBL C 93 "PI" 94 AVIEW 95 SF 01 96 XEQ 02 97 / 98 RCL 07 99 1/X 100 - 101 1/X	Schaltung: π-Glied, Anzeige/Druck, Rahmenprogramm



XEQ 'RO' SIZE010 USER

$R_a \uparrow R_b$ a_{soll} (T) a_{soll} (TT) → a_{min} → $R_{1;2;3}$

Stroinski
Sept. 80

```
102♦LBL 03
103 ADV
104 XEQ 04
105 "⊢1"
106 AVIEW
107 STO 05
108 RCL 00
109 RCL 02
110 FS? 01
111 /
112 FC? 01
113 *
114 RCL 07
115 FS? 01
116 1/X
117 -
118 FS? 01
119 1/X
120 XEQ 04
121 "⊢2"
122 AVIEW
123 STO 06
124 RCL 07
125 XEQ 04
126 "⊢3"
127 AVIEW
128 ADV
129 STOP
130♦LBL 04
131 FC? 55
132 STOP
133♦LBL 06
134 CLA
135 ARCL X
136 ARCL 08
137 RTN
138♦LBL 05
139 FC? 55
140 STOP
141 CLA
142 ARCL X
143 ARCL 09
144 END

SIZE 010
TOT. REG. 42
USER MODUS:
224 BYTES.
```

Anzeige/Druck von R_1

R_1 speichern

Berechnen von R_2

– nur π-Glied!

– nur T-Glied!

Anzeige/Druck von R_2

R_2 speichern

Anzeige/Druck von R_3

Anzeige/Druck-routine für R_b, R_1, R_2 und R_3

Anzeige/Druck-routine für R_a

Anzeige/Druck-routine für a_{min} → aM und a_{soll} → aS

Registerinhalte:

00	$\frac{n+1}{n-1}$
01	R_a
02	R_b
03	a_{min}
04	$n = P_a/P_b$
05	R_1
06	R_2
07	R_3
08	Text
09	Text

Dieses Programm läuft mit angeschlossenem Drucker automatisch ab.

Ohne Drucker werden die eingegebenen Werte mit Text einzeln quittiert. Nach jeder Anzeige stoppt das Programm und muß neu gestartet werden:

a) nach erfolgter Eingabe durch die entsprechende Labeltaste A, B oder C

b) zur Ausgabe des nächsten Wertes mit der R/S-Taste.

Das Anzeigeformat wurde mit Rücksicht auf ein geschlossenes Bild mit ENG 3 festgelegt. Mehrstellige Darstellung ist nach Aufruf der abgespeicherten Daten möglich.

Auswertung aussagenlogischer Ausdrücke $A=A(p_o,...,p_n)$ $(n \leq 9)$

J.Schu

A ist unter einem globalen Label zu speichern. Beim Aufruf von A steht p_i in Register i. Die Programmierung erfolgt nach den Regeln der UPN. Der Stack kann wie bei arithmetischen Ausdrücken verwendet werden. Folgende Funktionen sind vorhanden:

Name	Zchn	Register	X → LASTX	Bemerkung
AND	$\wedge$	X,Y	ja	
EQU	$\Leftrightarrow$	X,Y	ja	entspricht * (Mult.)
IMP	$\Rightarrow$	X,Y	ja	Y ⇒ X, Reihenfolge notwendig!
NOT	$\neg$	X	nein	entspricht CHS
OR	$\vee$	X,Y	ja	

Beispiel: $(p_2 \Rightarrow p_1) \vee (p_1 \wedge p_o)$ $(n=2)$

RCL o2 RCL o1 XEQ'IMP RCL o1 RCL oo XEQ'AND XEQ'OR

Das Programm wird mit XEQ'ASL gestartet. Auf AUSSAGE? ist das Label von A mit R/S einzugeben (Rechner hält im ALPHA-Modus). Auf VAR-ANZ? ist die Anzahl der Variablen (=n+1) mit R/S einzugeben. Danach wird zeilenweise die Tabelle der Wahrheitswerte ausgegeben: die Wahrheitswerte der p_i und - durch eine Leerstelle getrennt - der Wahrheitswert von A. Die Wahrheitswerte sind W und F. In der ersten Zeile haben alle p_i den Wert W, in der letzten F. Der Wert von p_o wechselt in jeder Zeile, der von p_1 in jeder zweiten, der von p_2 in jeder vierten etc. Die Tabelle hat 2↑(n+1) Zeilen. Die Ausgabe wird mit STOP beendet.

Zum Programm: Intern werden die Wahrheitswerte durch +1 und -1 dargestellt. Der Wahrheitswert von p_i wechselt in Zeile k (d.h. $p_i \rightarrow -p_i$), wenn gilt:

$$o = r_{ik} := (2^{n+1}+1-k) \bmod 2^i$$

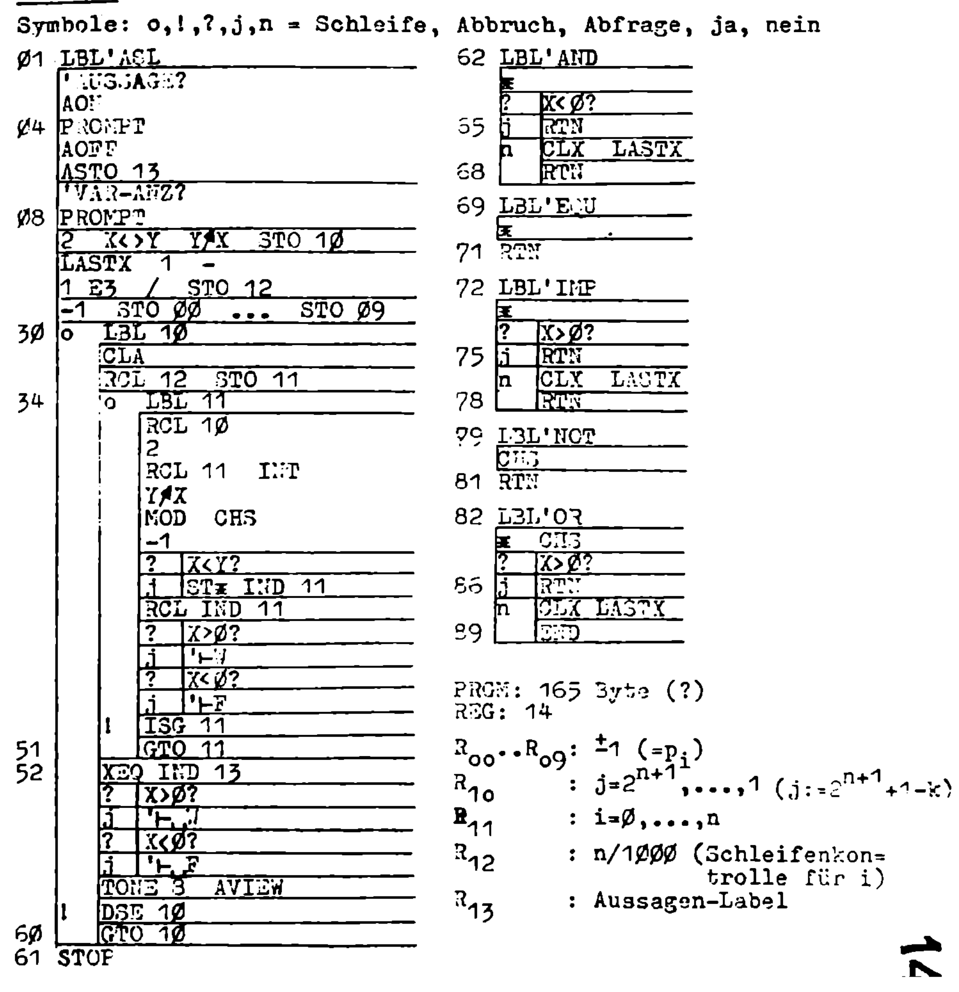
(dies gilt auch für die erste Zeile, da die p_i mit -1 vorbesetzt werden); dazu ist äquivalent:

$$-1 < -r_{ik} ;$$

in dieser Form erfolgt die Abfrage im Programm.

Hinweis: Ist man sich nicht sicher über die Wirkung einer Funktion, so kann man sich deren Tabelle ausgeben lassen, etwa: $p_o \Rightarrow p_1$, also: RCL oo RCL o1 XEQ'IMP.

mit TONE 8



Listing

Symbole: o,!,?,j,n = Schleife, Abbruch, Abfrage, ja, nein

```
Ø1 LBL'ASL
   'AUSSAGE?
   AON
Ø4 PROMPT
   AOFF
   ASTO 13
   'VAR-ANZ?
Ø8 PROMPT
   2  X<>Y  Y↑X  STO 1Ø
   LASTX  1  -
   1 E3  /  STO 12
   -1  STO ØØ  ...  STO Ø9
3Ø o  LBL 1Ø
      CLA
      RCL 12  STO 11
34    o  LBL 11
         RCL 1Ø
         2
         RCL 11  INT
         Y↑X
         MOD  CHS
         -1
         ?  X<Y?
         j  ST* IND 11
         RCL IND 11
         ?  X>Ø?
         j  '⊢W
         ?  X<Ø?
         j  '⊢F
      !  ISG 11
51       GTO 11
52    XEQ IND 13
      ?  X>Ø?
      j  '⊢W
      ?  X<Ø?
      j  '⊢F
      TONE 3  AVIEW
   !  DSE 1Ø
6Ø    GTO 1Ø
61 STOP

62 LBL'AND
   *
   ?  X<Ø?
65 j  RTN
   n  CLX  LASTX
68    RTN
69 LBL'EQU
   *
71 RTN
72 LBL'IMP
   *
   ?  X>Ø?
75 j  RTN
   n  CLX  LASTX
78    RTN
79 LBL'NOT
   CHS
81 RTN
82 LBL'OR
   *  CHS
   ?  X>Ø?
86 j  RTN
   n  CLX  LASTX
89    END
```

PRGM: 165 Byte (?)
REG: 14

$R_{00}..R_{09}$: ±1 (=p_i)
R_{10} : $j=2^{n+1},\dots,1$ ($j:=2^{n+1}+1-k$)
R_{11} : $i=\emptyset,\dots,n$
R_{12} : n/1ØØØ (Schleifenkontrolle für i)
R_{13} : Aussagen-Label

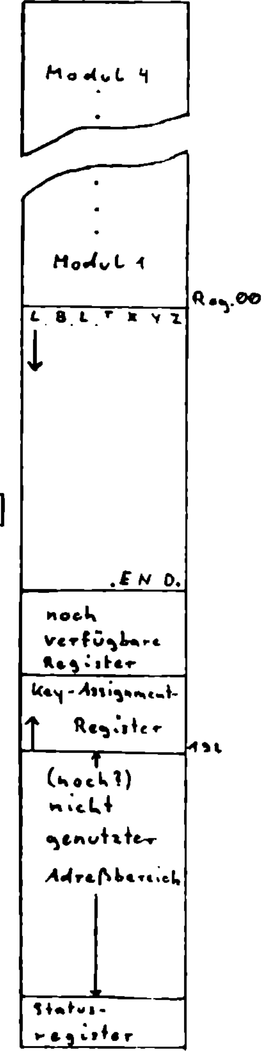
Glashütten, 19.4.1981

Liebe Clubmitglieder !
Auf der Hannovermesse bin ich durch Gespräche mit verschiedenen Mitgliedern in meinem Vorhaben bestärkt worden, in unregelmäßiger Folge erläuternde Artikel über Synthetisches und Rechnerorganisation zu schreiben. Ich halte dies für sinnvoll, da es einmal Mitglieder gibt, die, weil sie erst später in den Club eingetreten sind, manches verpaßt haben, zum anderen liegen schon sehr viele Kenntnisse und Erfahrungen anderer(meist PPC) vor, auf die man aufbauen sollte, anstatt sie sich mühsam selbst zuerarbeiten.

Ein paar Worte möchte ich noch über Unsinn bzw. Sinn der synthetischen Programmierung verlieren : Wer sich mit Synthetischem beschäftigt wird merken, daß es viele unsinnige, nutzlose Befehle gibt, aber auch sehr leistungsfähige. Zum einen lassen sich mit ihnen kurze und äußerst praktische Routinen herstllen wie SIZE-,Σ REG-,.END.-Finder, zum anderen ist häufig eine Verkürzung "normaler" Programme möglich, was dem Platzverbrauch und der Ausführungszeit zugute kommt.

Als erstes möchte ich mich dem Key-Assignment zuwenden, und zwar der Rechnerorganisation, als auch dem KA-Programm :
Wie manchen(oder vielen ?) schon bekannt ist, teilt sich der Speicherbereich des HP-41C folgendermaßen auf : siehe Skizze
Rechnerintern existiert eine absolute Addressierung nach Registern à 7 Bytes von 0 bis 255+nx64(n: Anzahl d. Module) und eine entsprechende innerhalb der Register von 0 bis 6 aufsteigend. Diese Addressierung erfolgt hexadezimal, bei vier Memory-Modulen(511) sind also maximal drei hexadezimale Ziffern(1FF) notwendig. Der dem Benutzer normalerweise zugängliche Speicherbereich liegt zwischen Register 192 und 255+nx64. Bei SIZE 000 beginnt das erste Benutzerprogramm bei der höchsten Addresse und läuft nahc unten fort. Die Programmzeilennummerierung ist also der absoluten entgegengesetzt. Die .END.-Marke begrenzt den Programmspeicher nach unten; sie ist nichts anderes als eine besondere END-Marke, die immer in den ersten drei Bytes eines Registers liegt. Gibt man einen Befehl ein in den Programmspeicher, so kontrolliert der Rechner erst, ob eine entsprechende Anzahl Nullcodes an dieser Stelle vorhanden ist, wenn ja, schreibt er den Befehl hinein, wenn nicht, schiebt er ein noch verfügbares Register ein. Durch diesen Einschub rutscht die .END.-Marke ein Register weiter nach unten. Hierdurch erklärt sich, daß sie im Druckerlisting bei Aus-

führung von Catalog 1 3 bis 7 Bytes verbraucht.
Ab Register 192 finden sich die Key-Assignmentregister, deren Aufbau im Weihnachtsinfo von Matthias Grabiak erklärt ist(entgegengesetzte Nummerierung der Bytes!). Die "jüngsten", vom Rechner ausgeführten Zuordnungen belegen immer Register 192, d.h. ältere wereden nach oben geschoben. Zwischen dem obersten KA-Register und dem Register, indem sich die .END.-Marke befindet, liegen die für den Benutzer noch verfügbaren Register, die immer gelöscht sind.



Hat man einen SIZE xyz gewählt, so ergibt sich die absolute Addresse des ersten Datenregisters (Reg.00) aus 256+nx64-xyz, die relative Datenregisternummer steigt also entspecbend der ab - soluten Addressierung an. Der Programmspeicher beginnt dann bei 255+nx64-xyz.
Im Statusregister c sind in jeweils drei hexadezimalen Ziffern die absoluten Addressen der Positionen der .END.-Marke, des ersten Datenregisters und des ersten Summenregisters abgespeichert : | ΣREG | | 1 6 9 | REG 00 | .END. |
Ein Tip am Rande : Speichert man in die Σ REG-Addresse drei Nullen, so werden bei Ausführung von CLΣ der gesamte Stack+Last X+Register M gelöscht. Diese Konfiguration ist durch folgende Befehlsfolge leicht zu erreichen : RCL c , STO M , APPEND"AA" , CLX , STO N , APPEND "AAAAA" , RCL N , STO c .
Damit sind Wir schon fast beim KA-Programm. Nach der Eingabe ermittelt das Programm die dem Tastencode entsprechende Dezimalzahl, wandelt diese und die anderen beiden in einen hexadezimalen Code um und setzt im Register ⊢(R) bzw. e das der Taste entsprechende Bit; ohne diese Information weiß der Rechner nicht, ob einer Taste etwas zugeordnet ist. Ist ein Bit

für eine Taste gesetzt ohne korrespondierende Information in einem der KA-Register so erscheint bei gedrückter Taste üblicherweise XROM 03,45 oder XROM 03,46(ABS,1/X). In besonderen Fällen kommt es vor, daß das Display erlischt und der Rechner für ca. 10 Sekunden wegtaucht. Durch ASN ALPHA ALPHA xy wird das Bit gelöscht.
Nach jeder geraden Anzahl von Eingaben wird ein kompletter 7-Bytes-Code entsprechend abgespeichert-wie, ist recht einfach. Da der Rechner seine Speicherhaushaltung über Register c ab wickelt, speichert man dort einfach die Information ab, daß das erste Datenregister bei Addresse 192, die .END.- Marke bei Addresse 191 liegt. Speichert man nun die sieben Bytes in Datenregister 00 , so gelangt der Code genau an die richtige Stelle. Die alte Information wird nun wieder nach Register c zurückgespeichert, so daß alles beim Alten ist, außer zwei neuen Zuordnungen.
Das Byte F0 muß immer in einem KA-Register stehen, da es dem Rechner als Erkennung dient. Auch darf nicht aus Versehen in das Register der .END.-Marke ein Assignment hineingespeichert werden. Um dies zu vermeiden, zu Beginn einfach GTO .. ausführen und in PGRM-Modus schalten. Die Meldung 00 REG xyz gibt an, daß zu den bereits existierenden KA-Registern noch xyz Register hinzukommen können.
Bei dieser Gelegenheit möchte ich ein von mir optimiertes KA-Programm vorstellen. Es benötigt nur eine Karte und hat den Vorteil, daß alte Zuordnungen nicht zwangsläufig überschrieben werden, sondern erhalten bleiben. Will man Befehle zuordnen, so muß eingetippt werden, wie viele KA-Register belegt sind bzw. bleiben sollen. Ist keines belegt, muß eine 0,!nicht CLX!eingegeben werden, da das Programm mit Flag 22-Abfrage testet, ob es die erste Zuordnung ist. Bei der Eingabe der belegten KA-Register ist zu beachten, daß der Rechenr, wenn man z.B. von vier Zuordnungen zwei gelöscht hat, diese sich aber nicht in demselben KA-Register befundeh haben, die restlichen zwei Zuordnungen nicht zusammenpackt also immer noch zwei KA-Register verbraucht sind. Nur wenn zwei Zuordnungen in demselben KA-Register gelöscht wurden, schiebt der Rechner sie nach PACKING zusammen.
Ist die Zahl eingetippt, erscheint nach XEQ"KA in der Anzeige KEY 1, worauf die gewohnten Eingaben gemacht, dann R/S gedrückt werden müssen. Nach 8 bis 10 Sekunden erscheint dann KEY 2 in der Anzeige. Nach der Eingabe und Druck auf R/S werden diese

beiden Zuordnungen endgültig abgespeichert und der Rechner erscheint im alten Status, alle Flags bleiben erhalten außer Flag 22, das gelöscht ist. Sollen weitere Zuordnungen gemacht werden, einfach auf R/S drücken, es erscheint KEY 3 usw. Auf keiner Fall darf eine numerische Eingabe erfolgen, wenn man weiter zuordnen möchte, da das Programm sonst die im X-Register stehende Zahl als die Anzahl zu überspringender Register auffaßt. Ist aus Versehen doch eine Eingabe erfolgt, dann CF 22, R/S ausführen. Durch diese Art des Programmablaufs kann beliebig weiter zugeordnet werden, ohne daß die Anzahl vorher festgelegt werden muß. Wird man durch Frau, Freundin oder Hund daran gehindert, die jeweils zweite Eingabe zu machen, so wird durch schnelles Tippen von R/S die erste Zuordnung korrekt abgespeichert, d.h. in diesem KA-Register ist dann nur ein Assignment. Möchte man nach vollbrachter Bändigung des Störenfrieds fotrfahren, muß nur die entsprechende Zahl von zu überspringenden KA-Registern eingegeben werden, um das halbgefüllte anzusprechen. Für ganz Eilige, die schon bei der jeweils ersten Eingabe gestört werden, besteht die Möglichkeit, durch R/S ihren alten Status wiederzuerlangen.

Wer das Programm in seinen einzelnen Schritten nicht versteht, sollte es ab Zeile 120 nicht im SST durchgehen. Für den Programmteil zwischen Zeile 50 und 71 ist unbedingt FIX 0 und CF 29 erforderlich (→ 60., → 71).

Es ist darauf zu achten, daß, wenn keine alten Zuordnungen vorhanden sind, eine 0 eingegeben wird. Stehen im Register 192 nur Nullcodes, so ignoriert der Rechner höhergelegene KA-Register. Da mein Programm etwas anders abläuft als das alte, darf auf keinen Fall eine 0 als erste Eingabe gemacht werden, was auch nicht sinnvoll ist, da die Codes 00 bis 0F als erstes Befehlsbyte die gleiche Wirkung haben, sie werden nämlich ignoriert, da ihnen außer bei mehrbytigen Befehlen nur die Platzhalterrolle zukommt.

So, das wär's zu dem Programm. Wer es auf Magnetkarte zugeschickt haben möchte, schicke mir bitte eine Magnetkarte und einen frankierten Brieumschlag.

Einen Appell noch an Euch : Ich möchte nicht, daß unser Club nur dazu dient, irgendweiche Programme auszutauschen, und die Action, sprich Neuentdeckungen sich nur in Amerika abspielen.

Wir sind genauso dazu in der Lage, siehe Memorysaver von ███ven, Wolf- und Goose-Befehl von Matthias,PBJ von Rolf. Natürlich läuft das bei uns in einem kleineren Rahmen, aber es soll auch weiterlaufen. Aus diesem Grund möchte ich versuchen, eventuell Unklares aufzuhellen und zuerläutern. Über Anregungen, Kritik und Fragen freue ich mich.

```
01♦LBL "KA      27 X<>Y       53 10↑X        79 R↑          105 FS? 07
02 FC?C 22      28 X<0?       54 RCL '       80♦LBL 02      106 "♦♦♦"
03 GTO 00       29 SF 01      55 FS? 01      81 2           107 ARCL 02
04 ABS          30 ABS        56 X<> e       82 /           108 RCL 01
05 STO 00       31 ENTER↑     57 STO [       83 FRC         109 ENTER↑
06 ,9           32 DSE X      58 DSE Y       84 X≠0?        110 RND
07 STO 01       33 10         59 ARCL Y      85 SF IND Y    111 2
08♦LBL 00       34 ST/ Z      60 X<> [       86 ISG Y       112 MOD
09 1            35 MOD        61 X<> d       87 X<> L       113 X=0?
10 X<> d        36 8          62 SF IND Z    88 INT         114 GTO 04
11 STO a        37 *          63 X<> d       89 X≠0?        115 -
12 FIX 0        38 ST+ Z      64 X<> [       90 GTO 02      116 2
13♦LBL 01       39 +          65 R↑          91 R↑          117 /
14 STO 02       40 INT        66 6           92 X<> d       118 RCL 00
15 "KEY "       41 +          67 LASTX       93 CLA         119 +
16 ARCL 01      42 36         68 -           94 STO [       120 RCL c
17 ISG 01       43 LASTX      69 10↑X        95 "├***"      121 RCL [
18 PROMPT       44 -          70 ARCL X      96 RCL \       122 "µ♦ "
19 FC?C 22      45 RCL X      71 X<> \       97 CLA         123 ASTO c
20 GTO 03       46 8          72 FS? 01      98 STO [       124 STO IND Z
21 256          47 FS? 01     73 STO e       99 ASTO X      125 RDN
22 ST* T        48 ST+ T      74 FC?C 01     100 FS?C 07    126 STO c
23 R↑           49 ST/ Z      75 STO '       101 GTO 01     127♦LBL 04
24 R↑           50 MOD        76 0           102♦LBL 03     128 RCL a
25 +            51 X<>Y       77 X<> d       103 ""         129 STO d
26 *            52 INT        78 -23         104 ARCL X     130 CLST
                                                            131 END
```

222 BYTES
SIZE 003
106 F4F0000000
122 F30C00BF

Ansonsten

Happy Programming

Andreas

Markscheffel (69)

...ung: Das Prgr ist bei Ralf Pfeifer und Walter Pieperhoff auf Karte zu haben!
Oliver

HP-41CV - Ein erster "E[illegible]nisbericht"

Nachdem ich meinen HP-41C mit einem Memory Module für 600,-DM verkauft hatte, bestellte ich mit dem Schein aus Prisma/Januar, unter Angabe der Mitgliedsnummer bei Fa. Hansen einen HP-41CV. Er kam schon in 5 Tagen und kostete nur 668,-DM!!
Die Besonderheiten des CV in Stichworten:

1) Eine neue, abgeflachte Tastatur. Eine wirkliche Verbesserung.
2) 319 Register bedürfen keines Kommentares.
3) Die Grundplatte, aus der die Tasten ragen ist aus rauhem Kunststoff. Sie verschmutzt und verkratzt nicht so leicht wie die alte Metallplatte.
4) Zu den üblichen deutschen Anleitungsbüchern liegt das "HP-41C Operating Manual- A Guide for the Experienced User" bei. Ein 72 Seiten starkes neues Anleitungsbuch (noch?) in englisch. Es enthält sowohl in komprimierter Form die Besonderheiten eines HP-41C, als auch als auch bisher nicht bekannte Informationen zum Rechner. Nicht zuletzt aufgrund seines Sachverzeichnisses ist es brauchbarer als die beiden anderen Bücher zusammen.
5) Der Fehler bei Alpha-Vergleichen (A=Y?; A≠Y?) ist endlich beseitigt.
6) Der wichtige Catalog-Bug (PPC-Bug 9) ist noch vorhanden. Ein paar andere auch: Scrolling, 2*[←] nach SIZE, DEL, o. äh. hängt den 41 auf, tan90° ≠ ∞)
7) Die Seitenklappe ist noch da; die Kontakte zum Stromanschluß fehlen. Wer jedoch extern Strom anlegen will, kann das mit Krokodilsklemmen an den Batterieanschlüssen.
8) Der Vorteil eines rundherum neuen Geräts (vor allem die Ports) ist nicht zu unterschätzen. Fa. Hansen gewährt 1 Jahr Garantie.
9) Der CV ist etwas schneller als ein normaler 41C (kann auch eine Exemplarstreuung sein). Dies ist am leichtesten an den nach oben verschobenen Tonfrequenzen zu erkennen.

Fazit: Verkauft Eure "alten Maschienen" und holt Euch für geringen Aufpreis den Superrechner HP-41CV!!!!

Rolf Mach (75)

Frank Altensen

.lgemeines Programm zu CHIP-Preisausschreiben
;.B.: 12/8Ø; 1+2/81 etc.) HP-41 C

.s Programm basiert darauf, daß eine codierte Nachricht
s mehreren verschiedenen Elementen zusammengesetzt ist
d die Anzahl der sinnvollen Kombinationen kleiner als
e Anzahl der möglichen Kombinationen ist. Da der Rechner
er nicht zwichen sinnvollen und unsinnigen Lösungen
terscheiden kann, erstellt er hierbei die Menge der
glichen Lösungen und überläßt die Auswahl dem Bediener.
e Rechenzeit erhöht sich mit der Anzahl der möglichen
mbinationen.

ogrammbedienung:

Programm starten:	(XEQ"CHIP)	N= ?
Eingabe der Anzahl der Lösungsmöglichkeiten pro Frage:	N (R/S)	K= ?
Eingabe der Anzahl der Fragen:	K (R/S)	KOMB.=(N↑K) ?
Jetzt die Lösungsbuchstaben, beginnend mit dem ersten Buchstaben der ersten Frage, eingeben. Der Rechner starte das Prgm. automatisch nach dem letzten Zeichen.	$ (R/S)	

Der Rechner gibt alle möglichen Kombinationsmöglichkeiten
für diese Art von Rätseln aus. Das Aussortieren der richtigen
Lösung bleibt dem Bediener überlassen.

tenspeicher:

- Anzahl der Lösungen pro Frage

- Anzahl der Fragen

- N*K

- Zähler

-Ø6 - Belegt

) bis R_{N*K+9} - Elemente (alphanum.)

*K+1Ø bis $R_{N*K+9+K}$ - Stellen der Elemente

ZE=K*(N+1)+9

gs: keine

els:
) numerische:
Ø1 bis Ø5 und Ø7
alphanumerisch:
"CHIP

erprogrammebenen: keine

ipherie: eventuell Memory Module und Drucker
Drucker ist zu empfehlen, da Ausgabe manschmal
sehr Zeitaufwendig.

Frank Altensen

Allgemeines Programm z[illegible]HIP-Preisausschreiben
(z.B.: 12/8∅; 1+2/81 etc.)

Beispiel: Chip-Preisausschreiben vom 1.1.1981:

Vier Fragen mit jeweils drei möglichen Antworten pro Frage.

```
1.                (XEQ"CHIP)        N= ?
        3         (R/S)             K= ?
        4         (R/S)             KOMB.=81
                                     ?
        Z         (R/S)              ?
        C         (R/S)              ?
        O         (R/S)              ?
        H         (R/S)              ?
        X         (R/S)              ?
        U         (R/S)              ?
        .           .                .
        .           .                .
        .           .                .
        D         (R/S)              ?
        E         (R/S)              ?
        P         (R/S)              ZHYD
                                     ZHYE
                                     ZHYP
                                     ZHXD
                                     ZHXE
                                     ZHXP
                                      .
                                      .
                                      .
                                      .
                                      .
                                     OUIE
                                     READY
```

Unter den gefundenen Lösungen gibt es nur vier, die auch eine echte Bedeutung haben; nämlich: ZUSE, CHIP, OXYD u. OXID. OXYD und OXID können ausgaschloßen werden, da Sie inhaltlich eher der Chemie als der Informatik zuzurechnen sind. Jetzt kann man entweder auslosen, oder man weis, wer das erste Buch über Logarithmen geschrieben hat.

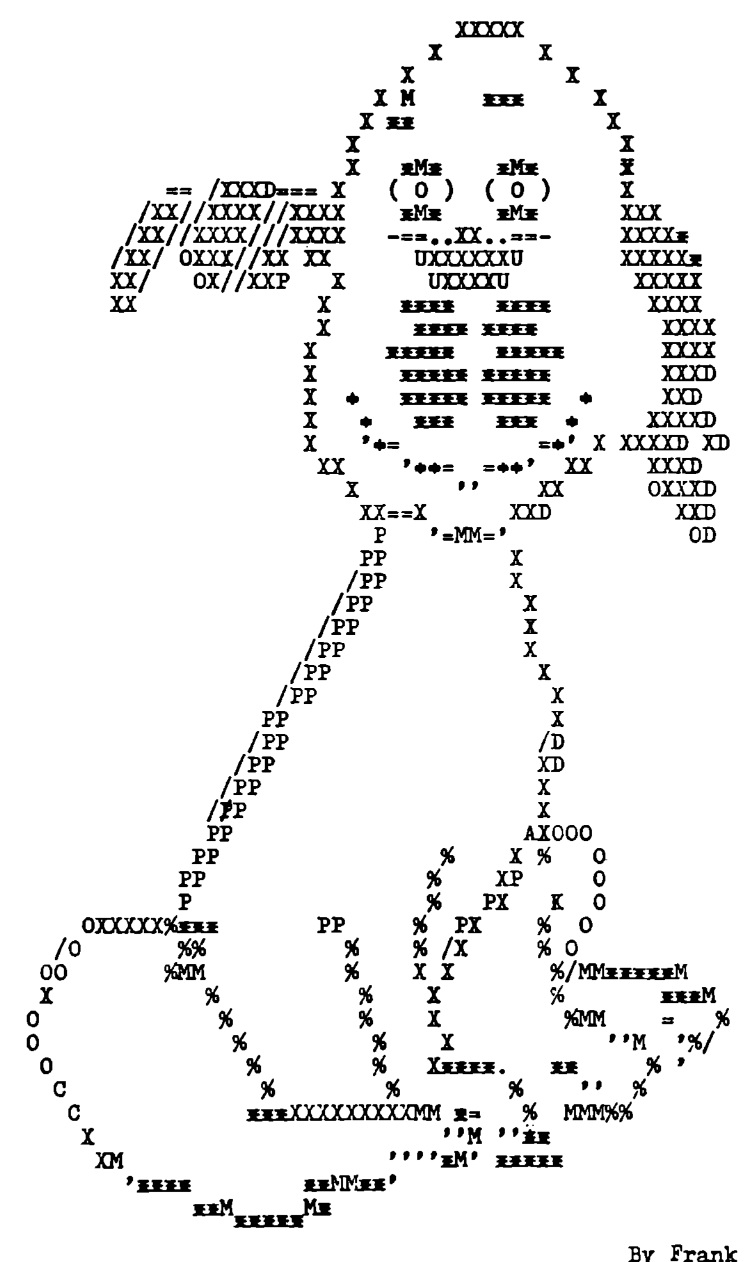
Viel Spass und Happy Programming

Frank Altensen

Frank Altens

Allgemeines Programm zu CHIP-Preisausschreiben
(z.B.: 12/8Ø; 1+2/81 etc.) HP-41 C

```
01 LBL"CHIP
02 CLRG
03 "N= ?
04 PROMPT
05 STO ØØ
06 "K= ?
07 PROMPT
08 STO Ø1
09 *
10 STO Ø2
11 "KOMB.=
12 RCL ØØ
13 RCL Ø1
14 Y/X
15 FIX Ø
16 ARCL X
17 AVIEW
18 PSE
19 FIX 9
20 RCL Ø2
21 1 E3
22 /
23 1Ø
24 ENTER/
25 ENTER/
26 1 E3
27 /
28 +
29 +
30 AON
31 1 E-3
32 -
33 LBL Ø1
34 " ?
35 PROMPT
36 ASTO IND X
37 ISG X
38 GTO Ø1
39 AOFF
40 RCL Ø2
41 1Ø
42 +
43 ENTER/
44 ENTER/
45 RCL Ø1
46 1
47 -
48 +
49 1 E3
50 /
51 +
52 STO Ø3
53 RCL ØØ
54 LBL Ø2
55 STO IND Y
56 ISG Y
57 GTO Ø2
58 LBL Ø4
59 RCL Ø3
60 FRC
61 1 E3
62 *
63 RCL Ø3
64 INT
65 1 E3
66 /
67 +
68 STO Ø6
69 1
70 RCL Ø1
71 1 E3
72 /
73 +
74 STO Ø5
75 RCL Ø3
76 STO Ø4
77 CLA
78 LBL Ø3
79 RCL IND Ø4
80 CHS
81 RCL Ø5
82 INT
83 RCL ØØ
84 *
85 +
86 1Ø
87 +
88 ARCL IND X
89 ISG Ø5
90 ADV
91 ISG Ø4
92 GTO Ø3
93 AVIEW
94 1Ø
95 STO Ø7
96 LBL Ø5
97 RCL IND Ø5
98 SIGN
99 X=Ø?
100 GTO Ø7
101 DSE IND Ø6
102 GTO Ø4
103 RCL ØØ
104 STO IND Ø6
105 DSE Ø6
106 ADV
107 DSE Ø7
108 GTO Ø5
109 LBL Ø7
110 "READY
111 BEEP
112 PROMPT
113 END
```



By Frank Altensen &
HP-983Ø

Programm: FN Flugnavigation

FN dient der vorgeschriebenen, navigatorischen Flugvorbereitung für Überlandflüge, sowie der Berechnung der Winkelkomponenten auf dem Start und Landeplatz.

1. Spuren 1 . . . 4 vom FN einlesen.

2. Programm mit XEQ FN starten.

3. Eingabe der Kennziffer n der ersten Flugteilstrecke (≤ 32)
   (Kennzifferbezeichnung der folgenden Flugteilstrecken erhöht sich automatisch um +1)
   Kartenkurs (°) und Entfernung (nm) der Flugteilstrecken im Dialog mit R/S eingeben, z.B. 150, 100 (Kartenkurs 150°, Entfernung 100 nm).

4. Eingabe der Flugdaten durch XEQ E
   Magnetische Variation (°), Geschwindigkeit - (nm) und Richtung (°) des Höhenwindes sowie der Geschwindigkeit über Grund (V_E in kn) des Flugzeugtyps im Dialog mit R/S eingeben.
   (Eingabe: Mag. Variation 3° West = -3
   3° Ost = +3)

5. Eingabe Kennziffer n der ersten gewünschten Flugteilstrecke mit R/S. Automatische Ausgabe der berechneten Navigationsdaten für die Flugteilstrecken beginnend mit der eingegebenen Kennziffer n.

	Kennziffer der Flugteilstrecke	
Automatische Ausgabe:	Kartenkurs	°
	Entfernung	nm
	Windabdriftwinkel	°
	Rechtweisender Windkurs	°
	Mißweisender Windkurs	°
	Geschwindigkeit über Grund	kn
	Flugzeit	Std, Min.
	Gesamtflugzeit	Std, Min.
	(nur bei letzter Flugteilstrecke)	

6. Berechnung der Windkomponenten durch XEQ F

   Eingabe der Start/Landebahnrichtung (°) der Richtung (°) und Geschwindigkeit (kn) des Bodenwindes im Dialog mit R/S.

Automatische Ausgabe der Gegenwind- und Rückenwind-Komponente in Knoten (Gegenwind positiv, Rückenwind negativ) und der linken oder rechten Seitenwindkomponente.
(Seitenwindkomponente von rechts positiv, Seitenwindkomponente von links negativ).

7. Datenkorrektur der vorigen Flugteilstrecken
   XEQ C. (ERROR-ANZEIGE, wenn Kennziffer < 1)

8. Datenkorrektur von Flugteilstrecken bei beendeter DATA-Eingabe
   XEQ B. Eingabe der Kennziffer mit R/S. Anzeige der betreffenden Flugteilstrecken - DATA Neueingabe mit R/S.

9. Abspeicherung der Flugteilstrecken - DATA auf KARTE XEQ D
   Erfolgt automatisch wenn Kennziffer größer 32 d.h. in der Regel, wenn mehr als 32 Flugteilstrecken vorliegen. Bei Ausführung von XEQ E (4) wird diese Karte zuerst im Programm für den ersten Rechendurchgang verlangt.

Vom Programm werden die Register R 00 . . . R 58 benutzt. Die eingelesenen DATA der Flugteilstrecken werden in den Speichern 1-32 gespeichert, Speicher 33-44 sind Rechenspeicher . In den Speichern 44-57 werden "strings" gespeichert.

Benötigt werden 2 Speichermodule.
Das Programm arbeitet mit und ohne Drucker.

x = Grad
nm = nautische Meilen
kn = Knoten

Liebe Clubfreunde !

Dieses recht umfangreiche Flugnavigationsprogramm wird hoffentlich bei allen Hobby-Flugbegeisterten auf Anklang stoßen, da es in der Flugvorbereitung die sonst verwendete Rechenscheibe weitgehend ablöst.
Mein besonderer Dank gilt Herrn Hoenow (11), der mir freundlicherweise bei der Optimierung des Programms behilflich war.

Happy Programming

Harald Wienbeck (141)
Rethwiese 13
2080 Pinneberg (Adressenänderung!!!)

```
 01+LBL "FN"
CF 01  SF 27  CLRG  CLX
XEQ 12  BEEP

 08+LBL 01
CLA  ARCL 45  ARCL 44
"⊢?"  PROMPT  STO 00  32
X<>Y  X>Y?  GTO 01

 19+LBL 02
RCL 00  33  X=Y?  GTO D
RDN  FIX 0  CLA  ARCL 46
"⊢ "  ARCL 44  ARCL X
AVIEW  STOP  FIX 3
VIEW X  STO IND 00  1
ST+ 00  GTO 02

 39+LBL 05
TONE 8  TONE 5  "WRONG "
ARCL 47  AVIEW  STOP

 46+LBL D
"WRITE LEGS"  AVIEW
1.032  WDTAX  SF 01  CLX
GTO "FN"

 54+LBL B
CLA  ARCL 47  ARCL 48
AVIEW  PSE  CLA  ARCL 45
ARCL 44  "⊢?"  PROMPT
STO 00  RCL IND 00
FIX 4  STOP  STO IND 00
STOP

 71+LBL C
1  ST- 00  RCL 00  X=0?
GTO 05  GTO 02

 78+LBL 03
RCL 00  32  X<>Y  X>Y?
GTO 04  RCL IND 00  0
X=Y?  GTO 04  RCL 00
FIX 0  CLA  ARCL 44
ARCL 46  "⊢ "  ARCL X
AVIEW  PSE  FIX 3
RCL IND 00  VIEW X
TONE 5  PSE  PSE  1
ST+ 00  GTO 03

106+LBL 04
"END "  ARCL 44  ARCL 46
AVIEW  TONE 5  TONE 6
STOP

114+LBL E
ADV  ADV  0  STO 40
FS? 01  XEQ 10  SF 12
CLA  ARCL 48  "⊢."
ARCL 46  "⊢:"  XEQ 11
CLA  ARCL 49  "⊢VARIA"
ARCL 54  "⊢ ?"  AVIEW
STOP  FIX 0  VIEW X
STO 33  CLA  ARCL 56
ARCL 57  "⊢ ?"  AVIEW
STOP  VIEW X  STO 34
CLA  ARCL 56  ARCL 55
ARCL 54  "⊢ ?"  AVIEW
STOP  VIEW X  STO 35
CLA  ARCL 51  "⊢AIR"
ARCL 57  "⊢ ?"  AVIEW
STOP  VIEW X  STO 36
RCL 33  X>Y?  GTO 06
CLA  ARCL 45  ARCL 44
"⊢?"  PROMPT  STO 00

173+LBL 07
RCL IND 00  X=0?  GTO 08
ADV  ADV  SF 12  CLA
ARCL 44  ARCL 00  XEQ 11
RCL IND 00  INT  STO 41
CLA  ARCL 51  "⊢TRACK "
ARCL 53  ARCL X  AVIEW
RCL IND 00  FRC  1 E3  *
STO 42  "DISTANCE"
ARCL 52  ARCL 53  ARCL X
AVIEW  RCL 41  CHS
RCL 35  +  STO 37  SIN
RCL 34  *  RCL 36  /
ASIN  FIX 1
"DRIFT ANGLE"  ARCL 53
ARCL X  AVIEW  RCL 41  +
STO 38  XEQ 09  FIX 0
CLA  ARCL 51  "⊢HEADING"
ARCL 52  ARCL X  AVIEW
RCL 33  -  XEQ 09  CLA
ARCL 49  "⊢HEADING"
ARCL 52  ARCL X  AVIEW
RCL 38  RCL 36  P-R
X<>Y  RCL 35  RCL 34
P-R  RDN  -  RDN  -  R↑
X<>Y  R-P  "GROUND"
ARCL 57  ARCL 53  ARCL X
AVIEW  STO 39  RCL 42
X<>Y  /  ST+ 40  HMS
FIX 2  CLA  ARCL 48
ARCL 50  ARCL 52  ARCL X
AVIEW  1  ST+ 00  FIX 0
GTO 07

275+LBL 06
TONE 0  TONE 5  TONE 9
CLA  ARCL 46  "⊢ ERROR"
AVIEW  GTO E

284+LBL 08
ADV  "END OF LEGS"
AVIEW  ADV  RCL 40  HMS
FIX 2  "TOTAL FL."
ARCL 50  ARCL X  AVIEW
ADV  ADV  STOP

299+LBL 09
360  MOD  RTN

303+LBL 10
CF 01  1.032  RDTAX  RTN

308+LBL F
SF 12  "TAKE OFF/"
AVIEW  PSE  "LANDING"
AVIEW  PSE  ADV  XEQ 13
FIX 0  CF 12  CLA
ARCL 55  ARCL 54  "⊢ ?"
AVIEW  STOP  VIEW X
STO 43  CLA  ARCL 56
ARCL 55  ARCL 54  "⊢ ?"
AVIEW  STOP  VIEW X
STO 41  CLA  ARCL 56
ARCL 57  "⊢ ?"  AVIEW
STOP  VIEW X  STO 42
ADV  RCL 43  RCL 41
RCL 33  -  -  CHS
RCL 42  P-R
"+HEAD/-TAIL"  ARCL 56
AVIEW  PSE  FIX 1
VIEW X  PSE  X<>Y
"+RIGHT/-LEFT"  AVIEW
PSE  "CROSS"  ARCL 56
AVIEW  PSE  VIEW X  ADV
ADV  ADV  ADV  RTN

375+LBL 11
AVIEW  "-------"  ASTO X
ARCL X  AVIEW  CF 12
ADV  RTN

384+LBL 12
"LEG "  ASTO 44
"NO OF "  ASTO 45
"DATA"  ASTO 46
"CORREC"  ASTO 47
"FLIGHT"  ASTO 48
"MAG. "  ASTO 49
"-TIME "  ASTO 50
"TRUE "  ASTO 51  "  "
ASTO 52  "    "  ASTO 53

405+LBL 13
"TION"  ASTO 54  "DIREC"
ASTO 55  "WIND"  ASTO 56
"SPEED"  ASTO 57  .END.
```

RECHEN-
BEISPIEL :

DATA LEG 1.
20.100
DATA LEG 2.
110.200
DATA LEG 3.
200.100
DATA LEG 4.
290.200
DATA LEG 5.

FLIGHT DATA:

MAG. VARIATION ?
-3.
WINDSPEED ?
15.
WINDDIRECTION ?
60.
TRUE AIRSPEED ?
95.

TAKE OFF/
LANDING

DIRECTION ?
110.
WINDDIRECTION ?
60.
WINDSPEED ?
10.

+HEAD/-TAILWIND
6.8
+RIGHT/-LEFT
CROSSWIND
-7.3

LEG 1.

TRUE TRACK	20.
DISTANCE	100.
DRIFT ANGLE	5.8
TRUE HEADING	26.
MAG. HEADING	29.
GROUNDSPEED	83.
FLIGHT-TIME	1.12

LEG 2.

TRUE TRACK	110.
DISTANCE	200.
DRIFT ANGLE	-6.9
TRUE HEADING	103.
MAG. HEADING	106.
GROUNDSPEED	85.
FLIGHT-TIME	2.21

LEG 3.

TRUE TRACK	200.
DISTANCE	100.
DRIFT ANGLE	-5.8
TRUE HEADING	194.
MAG. HEADING	197.
GROUNDSPEED	106.
FLIGHT-TIME	0.56

LEG 4.

TRUE TRACK	290.
DISTANCE	200.
DRIFT ANGLE	6.9
TRUE HEADING	297.
MAG. HEADING	300.
GROUNDSPEED	104.
FLIGHT-TIME	1.55

END OF LEGS

TOTAL FL.-TIME 6.26

L O H N S T E U E R (Österreich 1981)   16.1.1981 FI103

Programmzeilen 260   Systemkonfiguration: 1Memorymodul
Bytes 577   (Drucker)
Datenregister 002   (Kartenleser)

Das Programm dient zur Berechnung der österreichischen Lohnsteuer. Es ersetzt die Lohnsteuertabelle und kann für alle üblichen Lohnzahlungszeiträume verwendet werden. Das Programm kann als Unterprogramm in einem Lohnverrechnungsprogramm benutzt werden.
In das Programm eingearbeitet ist das Werbungskostenpauschale mit 4914,- und das Sonderausgabenpauschale mit 3276,-S.
Folgende Steuersätze sind dem Programm zugrundegelegt:

von	bis		
	50.000	23%	- 4800
50.100-	100.000	28%	- 7300
100.100-	150.000	33%	- 12300
150.100-	200.000	38%	- 19800
200.100-	240.000	43%	- 29800
240.100-	280.000	48%	- 41800
280.100-	320.000	52%	- 53000
320.100-	500.000	55%	- 62600
500.100-	1.000.000	58%	- 77600
000.100-	1.500.000	60%	- 97600
über 1.500.100		62%	-127600

Arbeitnehmerabsetzbetrag	3000,-
Alleinverdienerabsetzbetrag	3200,-
Pensionistenabsetzbetrag	2000,-

Eine Änderung der Steuersätze kann durch Überschreiben der betreffenden Programmzeilen leicht erfolgen. Dabei beachten, daß Betrag, Prozentsatz und Absetzbetrag in einer Zahl zusammengefaßt ist.
Wenn das Programm als Unterprogramm aufgerufen wird ist folgendes zu beachten:
Die Bemessungsgrundlage muß in R 01 gespeichert sein. Für tägliche Lohnzahlungen muß F 03, für monatliche Lohnzahlungen F 05 gesetzt sein. Für wöchentliche Zahlungen muß F 04 gesetzt sein, wobei die Anzahl Wochen mal 6 in R 00 gespeichert sein muß. Zur Berücksichtigung des Alleinverdiener-Absetzbetrages muß F 00 gesetzt sein.

Registerbelegung: 00 Steuertage
01 Bemessungsgrundlage

Flags: 00 mit Alleinverdienerabsetzbetrag
01 Pensionszahlung
02 Unterprogrammsteuerung
03 "
04 "
05 "
11 benutzt
21 benutzt
27 ON
29 benutzt

Walter Kropf
Schönowitzstraße 14
A-8700 Leoben

Example / Beispiel / Exemple / Esempio

a) Für folgende Monatslöhne ist die Lohnsteuer zu berechnen:
8461,50 mit AV-Absetzbetrag
21521,— " "
34231,35 ohne "

b) Bei täglicher Lohnzahlung ist die Lohnsteuer zu berechnen:
232,15 ohne AV-Absetzbetrag
327,10 mit "

c) Für folgende Wochenlöhne beträgt die Lohnsteuer:
1371,20 mit AV-Absetzbetrag
1841,17 ohne "

d) Bei 4wöchentlicher Lohnzahlung ist die Lohnsteuer zu Berechnen:
14186,30 mit AV-Absetzbetrag
19925,00 mit "
20100,— ohne "

Input	Output
a) XEQ'LSTEU' D 8461,5 R/S	BMGLG=8.461,50
Y R/S	MTL:AV:1.052,00
21521 R/S	BMGLG=21.521,00
Y R/S	MTL:AV 6.004,00
34231,35 R/S	BMGLG=34.231,35
N R/S	MTL:12.985,00
b) A B 232,15 R/S	BMGLG=232,15
N R/S	TGL 24,60
327,1 R/S	BMGLG=327,10
Y R/S	TGL:AV 41,00
c) A C 1 R/S 1371,2 R/S	BMGLG=1.371,20
Y R/S	1WTL:AV 80,20
1841,17 R/S	BMGLG=1.841,17
N R/S	1WTL:273,60
d) A C 4 R/S 14186,3 R/S	BMGLG=14.186,30
Y R/S	4WTL:AV:3.150,50
19925 R/S	BMGLG=19.925,00
Y R/S	4WTL:AV:5.568,00
20100 R/S	BMGLG=20.100,00
N R/S	4WTL:5.899,10

Walter Kropf
Schönowitzstraße 14
A-8700 Leoben

USER INSTRUCTIONS I
PROGRAMMABLAUF I
INSTRUCTIONS D'EMPLOI I
NORME OPERATIVE I

Instructions Operationen Instructions Istruzioni	Variables Dateneingabe Données Dati	Function(s) Taste(n) Touche(s) Tasti	Result Resultat Résultat Risultato
Programm einlesen, wenn Drucker ange-			
schlossen, Wahlschalter auf MAN, Start		XEQ'LSTEU'	' TG:WO:MO:MP'
Für tägliche Zahlungen nach Schritt 5			
für wöchentl. Zahlungen nach Schritt 8			
für monatl. Zahlungen nach Schritt 11			
für monatl.Pensionszahlg.nach Schritt 13			
für tägliche Lohnzahlung		B	'BMGL='
Eingabe der Bemessungsgrundlage	BMGL	R/S	'AV?'
wenn Alleinverdienerabs.B.berücksichtigt			
werden soll	Y	R/S	
wenn AV-Absetzb.nicht berücksichtigt			
werden soll	N	R/S	'TGL:nn,nn'
für nächste Berechnung für gleichen			
Zeitraum, nach Schritt 4		(R/S)	'BMGL='
für wöchentliche Zahlungen		C	'?WO'
Eingabe der Anzahl Wochen	Anz.Wochen	R/S	'BMGL='
weiter bei Schritt 4			
für monatliche Zahlungen		D	'BMGL='
weiter bei Schritt 4			
für monatl.Pensionszahlungen		E	'BMGL='
weiter bei Schritt 4			
für neue Rechnung mit anderem Lohn-			
zahlungszeitraum		A	' TG:WO:MO:MP'
weiter bei Schritt 2			
( ) nur erforderlich wenn kein Drucker			
verwendet wird.			

Walter Kr[illegible]
Schönowitzstraße 14
A-8700 Leoben

PROGRAM LISTING
PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

FI103

Line Zeile Ligne Linea	Keystrokes Tastenfolge Touches Tasti	Comments Kommentar Commentaires Commento	Line Zeile Ligne Linea	Key pressed Tastenfolge Touches Tasti	Comments Kommentar Commentaires Commento
01	•LBL "LSTEU"	Programmstart	55	CF 01	
02	SF 02	Flag 02 für Unterprgr.	56	•LBL 10	
03	FS? 03	Flag 03 für tägl.Lohnz.	57	26	
04	GTO B		58	STO 00	
05	FS? 04	Flag 04 für wöchentl.Zlg.	59	XEQ 01	Steuer berechnen
06	GTO 04		60	XEQ 03	Ergebnis runden
07	FS? 05	Flag 05 für monatl.Zlg.	61	CLA	
08	GTO D		62	FS? 01	
09	•LBL A	Aufruf als Hauptprgr.	63	"P"	
10	CF 02		64	"⊢MTL:"	
11	" TG:WO MO:MP"	Tastenzuordnung	65	XEQ 02	Ergebnis ausgeben
12	PROMPT		66	FS?C 02	
13	•LBL B		67	RTN	Rücksprung aus UProgramm
14	CF 01	tägliche Lohnzahlung	68	GTO 10	
15	1		69	•LBL 01	Bemessungsgrundlage
16	STO 00		70	"BMGLG="	eingeben
17	XEQ 01	Steuer berechnen	71	FC? 02	
18	1 E-4	Rundung des Ergebnisses	72	PROMPT	
19	-		73	FC? 02	
20	FIX 2		74	STO 01	
21	RND		75	ARCL 01	Bemessungsgrundlage drucken
22	"TGL:"		76	FS? 55	
23	XEQ 02	Ergebnis ausgeben	77	PRA	
24	FS?C 02	Rücksprung aus UProgramm	78	FS? 02	
25	RTN		79	GTO 09	
26	GTO B		80	•LBL 05	Alleinverdiener ?
27	•LBL C		81	"AV?"	
28	"?WO"	wöchentliche Lohnzahlungen	82	AON	
29	PROMPT	für wieviel Wochen	83	PROMPT	Y oder N
30	6		84	AOFF	
31	*		85	SF 06	
32	STO 00		86	ASTO X	
33	•LBL 04	Start für Unterprogramm	87	"Y"	
34	CF 01		88	ASTO Y	
35	XEQ 01	Steuer berechnen	89	X=Y?	SF 00 wenn yes
36	XEQ 03	Ergebnis runden	90	SF 00	
37	RCL 00		91	X=Y?	
38	6		92	CF 06	
39	/		93	"N"	
40	FIX 0		94	ASTO Y	
41	CF 29		95	X=Y?	
42	CLA		96	CF 00	CF 00 wenn no
43	ARCL X		97	X=Y?	
44	SF 29		98	CF 06	
45	RDN		99	FS? 06	Wiederholung bei falscher
46	"⊢WTL:"		100	GTO 05	Eingabe
47	XEQ 02	Ergebnis ausgeben	101	•LBL 09	
48	FS?C 02	Rücksprung aus UProgramm	102	RCL 01	Berechnung des Jahres-
49	RTN		103	RCL 00	einkommens
50	GTO 04		104	/	
51	•LBL E	monatliche Pensionszahlg.	105	312	
52	SF 01	SF 01 für Pensionszahlung	106	*	
53	GTO 10		107	4914	Werbungskostenpauschale
54	•LBL D	monatliche Lohnzahlung	108	FS? 01	xxxxxxxxxxxxxxxxxxxxxxx

Walter Kropf
Schönowitzstraße 1?
A-8700 Leoben

PROGRAM LISTING

PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

Keystrokes	Comments
109 CLX	=0 bei Pensionszahlung
110 -	
111 3276	Sonderausgabenpauschale
112 -	xxxxxxxxxxxxxxxxxxxxxxx
113 1 E-2	Rundung des Einkommens
114 -	
115 1 E2	
116 /	
117 FIX 0	
118 RND	
119 10	
120 /	
121 50,230048	Steuersatz bis 50,000
122 INT	
123 X<>Y	
124 X<=Y?	
125 GTO 06	
126 100,200073	Steuersatz bis 100.000
127 INT	
128 X<>Y	
129 X<=Y?	
130 GTO 06	
131 150,330123	Steuersatz bis 150.000
132 INT	
133 X<>Y	
134 X<=Y?	
135 GTO 06	
136 200,300198	Steuersatz bis 200.000
137 INT	
138 X<>Y	
139 X<=Y?	
140 GTO 06	
141 240,430290	Steuersatz bis 240.000
142 INT	
143 X<>Y	
144 X<=Y?	
145 GTO 06	
146 280,480418	Steuersatz bis 280.000
147 INT	
148 X<>Y	
149 X<=Y?	
150 GTO 06	
151 320,52053	Steuersatz bis 320.000
152 INT	
153 X<>Y	
154 X<=Y?	
155 GTO 06	
156 500,550626	Steuersatz bis 500.000
157 INT	
158 X<>Y	
159 X<=Y?	
160 GTO 06	
161 1000,580776	Steuersatz bis 1.000.000
162 INT	

Line / Key pressed	Comments
163 X<>Y	
164 X<=Y?	
165 GTO 06	
166 1500,600976	Steuersatz bis 1,500,000
167 INT	
168 X<>Y	
169 X<=Y?	
170 GTO 06	
171 ,621276	Steuersatz über 1.500.000
172 INT	
173 X<>Y	
174•LBL 06	
175 LASTX	Berechnung der Steuer
176 FRC	
177 1 E2	
178 *	
179 INT	
180 LASTX	
181 FRC	
182 1 E6	
183 *	
184 RDN	
185 *	
186 10	
187 *	
188 R↑	
189 -	
190 3000	Arbeitnehmerabsetzbetrag
191 -	xxxxxxxxxxxxxxxxxxxxxxxx
192 2000	Pensionistenabsetzbetrag
193 FC? 01	xxxxxxxxxxxxxxxxxxxxxxxx
194 CLX	
195 -	
196 3200	Alleinverdienerabsetzbetrg.
197 FC? 00	xxxxxxxxxxxxxxxxxxxxxxxxxx
198 CLX	
199 -	
200 1100	
201 X<>Y	
202 X>Y?	
203 GTO 08	Mindeststeuerbetrag über-
204 910	schritten
205 INT	Mindeststeuersätze
206 X<>Y	
207 X<=Y?	
208 GTO 07	
209 930,15	
210 INT	
211 X<>Y	
212 X<=Y?	
213 GTO 07	
214 960,2	
215 INT	
216 X<>Y	

Walter Kropfl
Schönowitzstraße 14
A-8700 Leoben

PROGRAM LISTING

PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

F1103

Line / Zeile / Ligne / Linea	Keystrokes / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commento	Line / Zeile / Ligne / Linea	Key pressed / Tastenfolge / Touches / Tasti

```
217 X<=Y?
218 GTO 07
219 998.3
220 INT
221 X<>Y
222 X<=Y?
223 GTO 07
224 1028.45
225 INT
226 X<>Y
227 X<=Y?
228 GTO 07
229 1068.65
230 INT
231 X<>Y
232 X<=Y?
233 GTO 07
234 .91
235 STO L
236*LBL 07          Mindeststeuersatz
237 LASTX
238 FRC
239 1 E3
240 *
241*LBL 08
242 312
243 /
244 RTN             Steuer pro Tag
245*LBL 03          Rundung des Ergebnisses für
246 RCL 00          nicht tägliche Steuer
247 *
248 1 E-3
249 -
250 FIX 1
251 RND
252 RTN
253*LBL 02          Ausgabe des Ergebnisses
254 FS? 00
255 "HAV:"
256 FIX 2
257 SF 21
258 ARCL X
259 AVIEW
260 END
LBL'LSTEU
END       577 BYTES
```

Printed line numbers in the second Line column: 51, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100; first column: 45, 50.

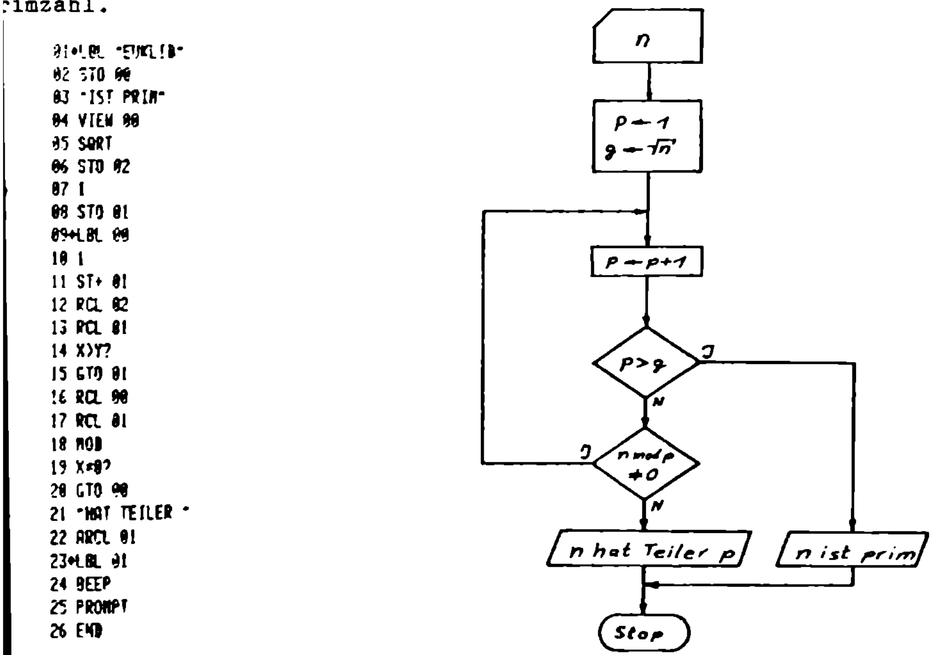
ei der Erstellung des weiter unten aufgelisteten Programms ging
s in erster Linie um möglichst kurze Rechenzeiten bei Zahlen mit
roßen Faktoren; dies erklärt die Länge des Programms von 693 Bytes.
ieses Programm soll weiterhin ein erster Schritt in Richtung
aktorisierung von Zahlen > 10^{10} in akzeptablen Zeiten (Tage?) sein.
afür ist dann sicher zusätzlicher Aufwand, wie z.B. programm-
esteuertes Einlesen von Daten oder Anschlußprogrammen notwendig.

ch habe nicht nur die Anregung, sondern auch viele Realisierungs-
ege dem Werk von D.E. Knuth (The Art of Computer Programming,
ol.II, Ch.4.5.4.) entnommen.

as Programm kombiniert zwei unterschiedliche Algorithmen, die in
rer Primitivform zunächst kurz vorgestellt werden sollen:

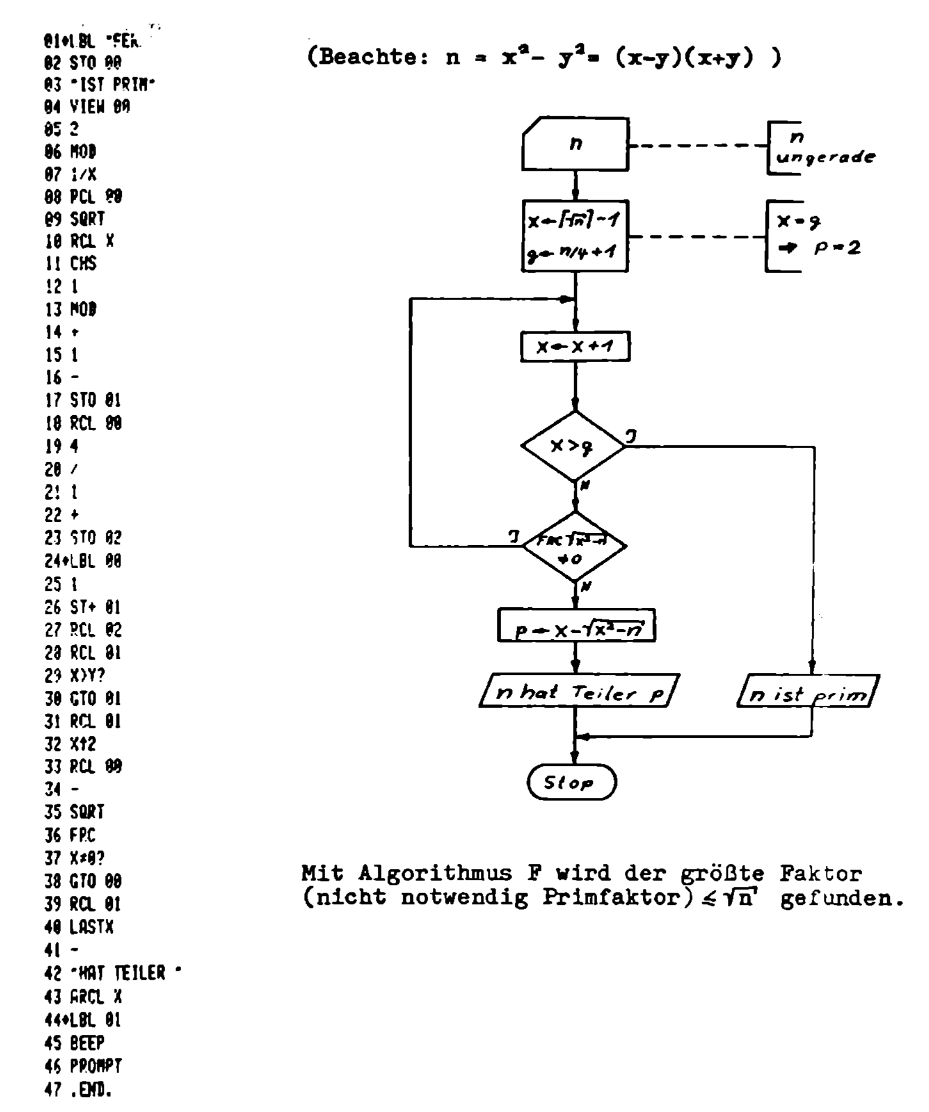
lgorithmus E (Die bekannte konventionelle Teilertestmethode; ich
ermute, daß sie auf Euklid zurückgeht (Die Elemente, neuntes
uch §14); wäre aber für Hinweise auf andere Quellen dankbar)

s wird nacheinander geprüft, ob 2, 3, 4, 5, 6, ... Teiler der
egebenen Zahl n sind. Der erste so erhaltene Teiler ist der kleinste
rimteiler von n. Ist bis $\sqrt{n}$ kein Teiler gefunden, so ist n eine
rimzahl.



orithmus F (Fermatsche Methode)

n die zu faktorisierende Zahl. Ist $n = F_1 \cdot F_2$ ($F_1 \leq F_2$), so kann
für F_1 und F_2 den Ansatz $F_1 = x-y$, $F_2 = x+y$ für geeignete
cht notwendig ganze) Zahlen x, y machen. Dann ist $x = (F_1+F_2)/2$,
$(F_2-F_1)/2$. Insbesondere sind x,y ganzzahlig, falls F_1 und F_2
de ungerade sind.
dem Fermatschen Algorithmus werden für ungerades n diese Werte
nd y gesucht: Ausgehend von $x = \lceil\sqrt{n}\rceil$ (kleinste ganze Zahl $\geq \sqrt{n}$)
d x erhöht und jeweils getestet, ob $x^2 - n$ eine Quadratzahl ist.




```
01•LBL "FER
02 STO 00
03 "IST PRIM"
04 VIEW 00
05 2
06 MOD
07 1/X
08 RCL 00
09 SQRT
10 RCL X
11 CHS
12 1
13 MOD
14 +
15 1
16 -
17 STO 01
18 RCL 00
19 4
20 /
21 1
22 +
23 STO 02
24•LBL 00
25 1
26 ST+ 01
27 RCL 02
28 RCL 01
29 X>Y?
30 GTO 01
31 RCL 01
32 X↑2
33 RCL 00
34 -
35 SQRT
36 FRC
37 X≠0?
38 GTO 00
39 RCL 01
40 LASTX
41 -
42 "HAT TEILER "
43 ARCL X
44•LBL 01
45 BEEP
46 PROMPT
47 .END.
```

(Beachte: $n = x^2 - y^2 = (x-y)(x+y)$)

Mit Algorithmus F wird der größte Faktor (nicht notwendig Primfaktor) $\leq \sqrt{n}$ gefunden.

Zur Charakterisierung der beiden Algorithmen seien nur zwei Beispiele mit den jeweiligen Rechenzeiten angegeben:

	E	F
1001423 = 887·1129	4 min 17 sec	4,8 sec
1001293 = 113·8861	33,5 sec	24 min 12 sec

Ist nur die Alternative zwischen Algorithmus E oder F gegeben, so entscheidet man sich ohne Zweifel für Algorithmus E (ca. $\sqrt{n}$ Tests gegenüber $n/4 - \sqrt{n}$ ($\gg \sqrt{n}$ bei großem n) Tests, zumal damit auf einfache Weise der Restfaktor weiter faktorisiert werden kann.
Beim Auswerten der gemessenen Rechenzeiten fällt aber doch etwas auf: Der Abstand vom Startwert ($\sqrt{n}$ bei F und 2 bei E) bis zum ersten gefundenen Faktor ist bei beiden Beispielen ungefähr gleich: $\sqrt{1001423} - 887 = 113,7 \approx 113$; d.h., es war zu erwarten, daß

Algorithmus F für 1001423 ungefähr die gleiche Zei[illegible] mehr, da der jeweilige Test aufwendiger ist) wie Algorithmus [illegible] für 1001293 benötigte. Aber offensichtlich ist hier der Fermatsche Algorithmus nahezu 7 mal so schnell. Das verwundert nicht, wenn man beachtet, daß in F mit der ersten Erhöhung von $x = \lceil\sqrt{n}\rceil = \lceil\sqrt{1001423}\rceil = 1001$ um 1 der "Testfaktor" $p = x-y$ einen Sprung von

$(x - \sqrt{x^2 - n}) - (x + 1 - \sqrt{(x+1)^2 - n}) = 25{,}8$ macht, während der Testfaktor im Algorithmus E konstant um 1 wächst. Die Erhöhung des Testfaktors $p = x-y$ wird mit wachsendem x allerdings immer geringer, was sich in der unverhältnismäßig hohen Rechenzeit beim Beispiel 1001293 ausdrückt. Man kann aber mit Recht erwarten, daß mit Algorithmus F große Faktoren wesentlich schneller gefunden werden als mit E.

Es liegt nahe, die beiden Verfahren zu kombinieren und den Übergang so zu wählen, daß eine optimale Rechengeschwindigkeit erreicht wird. Selbstverständlich wird zunächst Algorithmus E bis zu einem größten Testfaktor p_o angewandt, da damit schnell kleine Faktoren gefunden werden und somit in der Regel nur noch ein Restfaktor weiterfaktorisiert werden muß. Außerdem brauchen bei anschließender Anwendung von Algorithmus F nur solche x mit x-y zwischen $\sqrt{n}$ und p_o getestet zu werden. Die Berechnung eines optimalen Wertes p_o wird weiter hinten durchgeführt. Da dieser Wert immer größer als $\sqrt[3]{n}$ ist (wobei n der Restfaktor ist), kann n nur noch aus höchstens zwei Faktoren bestehen; somit liefert Algorithmus F dann auch in jedem Falle Primfaktoren.

Nun zur Realisierung der Algorithmen E und F im Programm. Analysiert man Algorithmus E, so fällt auf, daß es überflüssig ist, weiterhin gerade Faktoren p zu testen, wenn Teilbarkeit durch 2 nicht (mehr) gegeben ist. Damit wird die Anzahl der Tests praktisch halbiert; dementsprechend auch die Rechenzeit. Die entsprechende Überlegung gilt für den Testfaktor 3. Ist die zu faktorisierende Zahl nicht (mehr) durch 3 teilbar, so reicht es, den Testfaktor von 5 ausgehend abwechselnd um 2 oder 4 zu erhöhen. Damit werden alle Vielfachen von 2 und 3 übersprungen.
Eine Verminderung der Teilertests auf diese Weise läßt sich mit den Primfaktoren 5, 7, 11, ... fortführen. Jedoch zeigt die letzte Zeile der folgenden Tabelle, daß der zusätzliche Nutzen (Einsparung an überflüssigen Tests) immer geringer wird, während der Aufwand unverhältnismäßig ansteigt.

	Primitivversion a)	b)	c)	d)	e)	f)	g)	allgem.
Testfaktoren enthalten keine Primteiler bis einschließl.		2	3	5	7	11	13	p
Bereich, den eine Schleife überdeckt	1	2	6	30	210	2310	30030	$\Pi = 2 \cdot 3 \cdot 5 \cdot \ldots \cdot p$
Anzahl der Tests pro Schleife (Programmlänge)	1	1	2	8	48	480	5760	$\varphi(\Pi)$
Bereich, der mit einem Test überdeckt wird	1	2	3	3,75	4,38	4,81	5,21	$\Pi / \varphi(\Pi)$
Gewinn gegenüber vorherigem Fall		100 %	50 %	25 %	16,8 %	9,8 %	8,3 %	

(Dabei ist φ die Eulersche φ-Funktion: $\varphi(n)$ = Anzahl der zu n teilerfremden Zahlen kleiner n)
Da für jeden Test mindestens ein Unterprogrammaufruf mit Eingangsparameterübergabe notwendig ist, wären bei f) mindestens

480 x 3 = 14[illegible]tes nur für die Schleife notwendig (dabei wurde der relativ la[illegible]e indirekte Unterprogrammaufruf zugrunde gelegt). Bei e) sind mindestens 144 Bytes in der Schleife notwendig, was durchaus realisierbar ist. Für das folgende Programm wurde die Alternative e) zu d) untersucht, wobei allerdings in d) die Unterprogramme abgerollt wurden, d.h. nur im Erfolgsfall (Teiler gefunden) wird hier ein Unterprogramm aufgerufen (in diesem Fall ist der indirekte Unterprogrammaufruf nicht nur jeweils um ein Byte kürzer, sondern auch etwas schneller (d.h. das Überspringen ist schneller)).
Es zeigte sich, daß dieses Vorgehen nach d) die günstigsten Rechenzeiten lieferte. (Ein Schleifendurchlauf Zeile 052 bis 110 dauert 1,160 sec)

Realisierung des Fermatschen Algorithmus.
Während alle mir bekannten Realisierungen des Algorithmus E mindestens die Inkrementierung nach b) der obenstehenden Tabelle benutzen, ist es nicht ganz so offensichtlich, daß man im Algorithmus F die Anzahl der Tests ebenfalls (sogar effektiver) drastisch reduzieren kann.
Dieser Algorithmus wurde von Fermat benutzt, und es ist nicht anzunehmen, daß dieser sich für ein Verfahren entschied, bei dem er wesentlich häufiger die Wurzel aus einer Zahl ziehen mußte, als im anderen Falle eine Division durchzuführen gewesen wäre (zumal Fermat bekanntlich nicht einmal einen TI-Rechner zur Verfügung hatte, geschweige denn eine wirklich leistungsfähige Hilfe wie den HP-41C).
Fermats Vorgehensweise bestand darin, daß er die beiden Endziffern des Radikanden $x^2 - n$ betrachtete, um so viele Fälle auszuschließen; er arbeitete modulo 100. Soll nämlich $x^2 - n$ ein Quadrat sein, so müssen die Endziffern 00, a1, a4, 25, b6 oder a9 sein, wobei a gerade und b ungerade ist. 100 wurde sicher deshalb als Modul gewählt, weil man n mod 100 bei dezimal dargestelltem n direkt ablesen kann. Für die Realisierung mit dem Rechner ist das kein Vorteil mehr; daher kann man einen günstigeren Modul suchen (Im Programm habe ich 144 gewählt).
Um zu veranschaulichen, wie diese Vorgehensweise im Algorithmus F benutzt werden kann, sei dieses zunächst mit dem Modul 9 verdeutlicht: Voraussetzung ist, daß n schon soweit reduziert wurde, daß 3 kein Teiler mehr ist.
In der folgenden Tabelle sind in der Kopfspalte alle in Frage kommenden Reste n mod 9 aufgeführt (das sind $\varphi(9) = 6$ Stück, denn 0, 3 und 6 scheiden aus, da n nicht durch 3 teilbar ist).
In der Kopfreihe sind die Werte x mod 9 aufgeführt; das sind natürlich 9 Stück. Die Eintragungen in der Tabelle sind jeweils die Werte $x^2 - n$ mod 9, wobei diese Werte mit doppelter Zeichenbreite gedruckt wurden, falls sie ein Quadrat mod 9 darstellen. (Als Quadrate mod 9 kommen nur die Zahlen 0, 1, 4 und 7 in Frage, wie man leicht nachrechnen kann).

X↑2-N MOD 9 / N MOD 9	X MOD 9: 0	1	2	3	4	5	6	7	8
1	8	0	3	8	6	6	8	3	0
2	7	8	2	7	5	5	7	2	8
4	5	6	0	5	3	3	5	0	6
5	4	5	8	4	2	2	4	8	5
7	2	3	6	2	0	0	2	6	3
8	1	2	5	1	3	3	1	5	2

Es ist im Algorithmus F also beispielsweise uns [illegible] einen x - Wert zu testen, für den gilt: x mod 9 = 3 und gleichz[illegible]g n mod 9 = 4, denn der zu untersuchende Radikand x^2- n ist keine Quadratzahl, da er bei Teilung durch 9 den Rest 5 ergibt, und 5 ist kein Quadrat mod 9. Unter Berücksichtigung der Ergebnisse dieser Tabelle ergeben sichfolgende Inkremente für x:

n mod 9 = 2, 5 oder 8:
  $x \leftarrow \lfloor \sqrt{n}/9 \rfloor \cdot 9$ und dann jeweils um 3 erhöhen.

n mod 9 = 1:
  $x \leftarrow \lfloor \sqrt{n}/9 \rfloor \cdot 9 + 1$ und dann abwechselnd um 7 und 2 erhöhen.

n mod 9 = 4:
  $x \leftarrow \lfloor \sqrt{n}/9 \rfloor \cdot 9 + 2$ und dann abwechselnd um 5 und 4 erhöhen.

n mod 9 = 7:
  $x \leftarrow \lfloor \sqrt{n}/9 \rfloor \cdot 9 + 4$ und dann abwechselnd um 1 und 8 erhöhen.

Bei den ersten zwei (bzw. 3) Inkrementierungen muß zusätzlich abgefragt werden, ob x^2- n negativ ist.
Bei dieser Inkrementierung werden also mit einem Test mindestens 3, meistens sogar 4,5 x-Werte abgedeckt, im Gegensatz zu nur einem bei der Primitiv-Version.
Wie man sieht, sind jedoch leider die Inkremente abhängig vom Input (das ist beim Algorithmus E nicht der Fall). Die Berechnung der Inkremente erfolgt aber nur einmal und hat deshalb keine Rechenzeitverlängerung (im späteren Programm nur ca. 2 sec) zur Folge, sie verlängert das Programm jedoch erheblich.
Wie schon erwähnt wurde im Programm eine Inkrementierung mod 144 gewählt. Die der oben entsprechende Tabelle ist fast 1 m^2 groß, so daß sie hier aus naheliegenden Gründen nicht abgedruckt werden kann. Die Auswertung dieser Tabelle (und die anderer Module zu Testzwecken) erfolgte selbstverständlich durch ein Programm, dessen Arbeitsweise hier kurz geschildert sei (mit der Hoffnung, daß jemand die Abhängigkeit der Inkremente vom Input und vom gewählten Modul besser durchschaut als ich; das entsprechende Programm stelle ich als Listing oder auf 2 Magnetkarten gerne zur Verfügung).

Eingegeben wird lediglich der zu untersuchende Modul (hier 144). Nach geraumer Rechenzeit werden dann alle Quadrate modulo 144 ausgedruckt, und anschließend für jede zum Modul teilerfremde Zahl ein "Zettel" der folgenden Form:

```
------------

   144      ***

R10=   0
R11=   1
R12=   4
R13=   9
R14=   16
R15=   25
R16=   36
R17=   49
R18=   52
R19=   64
R20=   73
R21=   81
R22=   97
R23=   100
R24=   112
R25=   121
```

```
N MOD 144 = 109
N MOD 2 = 1
N MOD 72 = 37
N MOD 3 = 1
N MOD 48 = 13
N MOD 4 = 1
N MOD 36 = 1
N MOD 6 = 1
N MOD 24 = 13
N MOD 8 = 5
N MOD 18 = 1
N MOD 9 = 1
N MOD 16 = 13
N MOD 12 = 1

ANFANGSINCREMENT:

      1        ***

SCHLEIFENINCREMENTE:

     16        ***
     38        ***
     16        ***
      2        ***
     16        ***
     38        ***
     16        ***
      2        ***
```

```
N MOD 144 = 133
N MOD 2 = 1
N MOD 72 = 61
N MOD 3 = 1
N MOD 49 = 37
N MOD 4 = 1
N MOD 36 = 25
N MOD 6 = 1
N MOD 24 = 13
N MOD 8 = 5
N MOD 18 = 7
N MOD 9 = 7
N MOD 16 = 5
N MOD 12 = 1

ANFANGSINCREMENT:

      5        ***

SCHLEIFENINCREMENTE

      8        ***
     46        ***
      8        ***
     10        ***
      8        ***
     46        ***
      8        ***
     10        ***
```

Diese 48 [illegible]44)) "Zettel" wurden dann mühsam in die Programm-sequenz 122 [illegible] 234 (ohne 220 bis 229) umgesetzt, wobei ich sicher bin, daß man das eleganter machen kann.
An dieser Stelle sei eine zusätzliche Schwierigkeit genannt, die Algorithmus F bereitet: Da im Ausdruck $x^2 - n$ der x^2-Wert bei großem n größer als 10^{10} wird, ist keine korrekte Berechnung mehr möglich, wenn wie im obenstehenden Primitivprogramm inkrementiert wird. (Man versuche damit $98947 \cdot 101063$ zu faktorisieren).
Diese Schwierigkeit läßt sich aber umgehen, indem man ausnutzt, daß $(x + i)^2 - n = (x^2 - n) + (i^2 + 2ix)$ ist, wie das im späteren Programm realisiert wurde.

Im Gegensatz zu Algorithmus E, bei dem erhöhter Aufwand immer geringeren zusätzlichen Nutzen bietet, führt beim Algorithmus F die Wahl eines geeigneten größeren Moduls (mit mehr Primzahlpotenzen) zu wesentlich besseren Ergebnissen. Nach Knuth ist beispielsweise bei Berücksichtigung der ersten 30 Primzahlen (mit dem Taschenrechner natürlich nicht realisierbar) nur noch einer von 2^{30} Fällen zu testen.
Die Arbeitsweise von Programmen, die auch Zahlen $> 10^{10}$ faktorisieren, stelle ich mir so vor, daß beim Übergang zum Algorithmus F das Programm stoppt und dann nur noch anzeigt, welche von möglicherweise sehr vielen Datenkarten mit den richtigen Inkrementen eingelesen werden muß, da das Berechnen der Inkremente im Programm zwar nicht zu zeitaufwendig wäre, jedoch den Programmumfang so stark aufblähen würde, daß für die eigentlichen Testroutinen, die dann ja wesentlich komplizierter sind, kaum Platz bliebe.
Bei unserem Beispiel, dem Modul 144 werden je nach Beschaffenheit des Restfaktors n entweder 72 (n mod 3 = 1) oder 48 (n mod 3 = 2) x-Werte mit 4 Tests überdeckt. Das bedeutet, daß bei n in der Größenordnung 10^{10} mit den ersten 4 Tests 3720 bzw. 3050 Testfaktore p überdeckt werden. Dieser Bereich für p nimmt mit wachsendem x bis auf ca. 44 in beiden Fällen ab; bis dahin ist die Testmethode nach Algorithmus E schneller.

Zum Abschluß sei noch die Vorgehensweise erläutert, mit der die hoffentlich optimalen Übergangsstellen von E nach F gefunden wurden. Die "zeitfressenden" Schleifen im Programm sind die Programmsequenze 052 bis 110 für Algorithmus E und 262 bis 324 für Algorithmus F. Die Schleife für E wird bei meinem Rechner in $\alpha = 1{,}160$ sec durchlaufen, während für die Schleife in F die Zeit $\beta = 1{,}669$ sec benötigt wird. (Bei der folgenden Konstantenberechnung geht letztendlich nur das Verhältnis β/α ein, so daß unterschiedliche Rechnerlaufzeiten kompensiert wären).
Insgesamt muß vom Programm jeder in Frage kommende Faktor p im Bereich 0 bis $\sqrt{n}$ getestet werden, wobei das in folgender Reihenfolge geschieht:
Von 0 bis $k_m \cdot \sqrt{n}$ mit Algorithmus E und anschließend von $\sqrt{n}$ bis $k_m \cdot \sqrt{n}$ mit Algorithmus F; dabei ist m = n mod 3 (= 1 oder 2) und es ist $0 < k_1 < k_2 < 1$. Es sei weiter $B_m = 72$ oder 48 (für m = 1 oder 2).
Im Algorithmus F gilt:

$$n = p \cdot q = (x-y)(x+y) = x^2 - y^2$$

d.h. $p(x) = x - \sqrt{x^2 - n}$ bzw. $x(p) = (p^2 - n)/2p$

Gesucht wird nun zunächst eine Funktion f(p,n), die die Differenz des Verhältnisses der überstrichenen Testfaktorbereiche zu dem Verhältnis der Rechenzeiten für diese Bereiche angibt.

$$f(p,n) = \frac{p(x) - p(x - B_m)}{30} - \frac{\beta}{\alpha}$$

$$= \left(\sqrt{\left(\frac{n + p^2}{2p} + B_m\right)^2 - n} + \frac{p^2 - n}{2p} - B_m \right) \cdot \frac{1}{30} - \frac{\beta}{\alpha}$$

Die Faktoren k_n werden dann durch lineare Approximation der Nullstellen p dieser Funktion in Abhängigkeit von n ermittelt. Die Nullstellen dieser Funktion für verschiedene n-Werte (mit stärkerer Gewichtung großer Werte) wurden mit SOLVE (bzw. SOL) des Mathe I Moduls berechnet. Die lineare Approximation dieser Ergebnisse mit dem Kurvenanpassungsprogramm der Standardprogrammsammlung ergibt für
n mod 3 = 1: R^2=1,00000; a = 21,79642; b = 0,48023
und für
n mod 3 = 2: R^2=1,00000; a = 21,74261; b = 0,55692
Die absoluten Glieder a können vernachlässigt werden, da die Schleife in E bei einem Durchlauf schon 30 p-Werte überdeckt. Es ergibt sich also für das Programm:

k_1= 0,480 und k_2= 0,557

Die längste Rechenzeit ergibt sich für p = 10^{10} − 71 (größte Primzahl p < 10^{10} mit p mod 3 = 2): 45 min 46 sec, dagegen für 10^{10}−33 (größte Primzahl < 10^{10}): 41 min 23 sec.
Ein "ungünstiger" Fall ist sicher 55697·179533 mit 45 min 38 sec, denn hier würde der kleinere Faktor mit Algorithmus E direkt gefunden, wenn nicht zu F übergegangen würde.
55681·179591 mit 35 min 19 sec liegt kurz vor diesem Übergang. Diese "Ungereimtheit" liegt einfach daran, daß mit Algorithmus F von der Wurzel an abwärts getestet wird; dafür wird dann ja auch beispielsweise 99991·100003 in 30 min 31 sec gefunden.
(alle Rechenzeiten ohne angeschlossenen Printer).

Diese Beispiele zeigen, daß mit Algorithmus F für 52% des gesamten Testbereiches nur 26,3 % bzw. für 44,3 % nur 22,8 % der Gesamtrechenzeit benötigt werden.

Die Rechenzeit für das untenstehende Programm, das die Zahlen unterhalb 2^{19}, die die größten 10 Primzahlen < 2^{19} enthalten, faktorisiert, beträgt knapp 50 min.

```
01*LBL "P"
02 CF 26
03 88
04 STO 17
05 2
06 ENTER↑
07 19
08 Y↑X
09 87
10 -

11*LBL 01
12 XEQ "PRM"

13 RCL 00
14 1
15 +
16 DSE 17
17 GTO 01
18 SF 26
19 BEEP
20 END
```

```
524201
IS PRIME

524202
=2*3*7↑2
*1783

524203
IS PRIME

524204
=2↑2*29*4519

524205
=3↑3*5*11
*353

524206
=2*262103

524207
=113*4639

524208
=2↑4*3*67
*163

524209
=7*74[...]
```

```
5242[..]
=7*74897

524280
=2↑3*3*5*17
*257

524281
=269*1949

524282
=2*11*23831

524283
=3*174761

524284
=2↑2*131071

524285
=5*23*47*97

524286
=2*3↑3*7*19
*73

524287
IS PRIME

524288
=2↑19
```

EUROPÄISCHE BENUTZER-PROGRAMMBIBLIOTHEK
BIBLIOTHÈQUE EUROPÉENNE DE PROGRAMMES UTILISATEURS
LIBRERIA EUROPEA DEGLI UTILIZZATORI

HP 41C PROGRAM SUBMITTAL FORM

PROGRAMMFORMBLATT/DOCUMENTATION DU PROGRAMME/GENERALITÀ SUL PROGRAMMA

Program Title / Programmname / Titre du programme / Titolo del programma: PRIMFAKTORZERLEGUNG

Category No. / Kategorie Nr. / Catégorie N° / Categoria N°: 201
Name / Rubrik / Rubrique / Nome della categoria: Zahlentheorie

No. of program lines / Anzahl Programmzeilen / Nombre de lignes de programme / N° di linee di programma: 502

No. of data registers / Anzahl der benötigten Datenspeicher / Nombre de registres de données / N° di registri utilizzati: 116 (total)

Recommended HP 41C System configuration / Empfohlene System-Konfiguration / Configuration recommandée / Configurazione raccomandata
Port # 1: Memory Modul
Port # 2:
Port # 3: (Printer)
Port # 4:

This program requires the following programs as subroutines:
Dieses Programm benutzt folgende Programme als Unterprogramme:
Ce programme utilise les programmes suivants comme sous-programmes:
Questo programma usa i seguenti programmi come subroutines:

HP Applications ROM / HP Applikations ROM / ROM d'application HP / ROM di applicazione HP:
Program Name: / Programm: / Nom du programme: / Programma:

Program Abstract / Kurzbeschreibung / Résumé / Breve descrizione del programma

Durch die Kombination von konventionellem Testverfahren und der Fermat'schen Faktorisierungsmethode werden mit dem Programm ganze Zahlen n im Bereich $2 \leq n \leq 10^{10}$ schnell faktorisiert.

Die längste Rechenzeit ist ca. 46 min für die Primzahl $10^{10}-71$. $99991 \cdot 100003$ wird in ca. 31 min faktorisiert.

Name / Name / Nom / Nome: Friedrich Hillebrandt

Address / Anschrift / Adresse / Indirizzo: Jülicher Straße 443

City / Ort / Localité / Città: Aachen
Postal Code / Postleitzahl / Code postal / C.A.P.: 5100
Country / Land / Pays / Paese: Deutschland

ACKNOWLEDGMENT AND AGREEMENT

Erklärung und Ermächtigung/Déclaration et Autorisation/Dichiarazione e Autorizzazione

To the best of my knowledge, I have the right to contribute this program material without breaching any obligation concerning r or confidential information of other persons or organizations. I am contributing this program material on a nonconfidential nonob to Hewlett-Packard S.A. ("HP") for inclusion in its program library, and I agree that HP may use, duplicate, modify publish, program material, and authorize others to do so without obligation or liability of any kind. HP may publish my name and ad contributor, to facilitate user inquiries pertaining to this program material.

[illegible]

[illegible]

[illegible]

Date / Datum / Date / Data: 01.03.81

Signature / Unterschrift / Signature / Firma:

PROGRAMMABLAUF I
INSTRUCTIONS D'EMPLOI I
NORME OPERATIVE I

Step	Instructions	Variables	Function(s)	Result
1	Programm eingeben *			
2	Faktorisierung von n	n	XEQ "PRM"	n
				$= p_0{\uparrow}e_0 * p_1{\uparrow}e_1 \ldots$
				⋮
				oder
				IS PRIME
3	Faktorisierung von n	n	SF 00	
	mit Einzelanzeige der		XEQ "PRM"	n
	Primzahlpotenzen			$= p_0{\uparrow}e_0$
				$* p_1{\uparrow}e_1$
				⋮
				oder
				IS PRIME
	Nach Beendigung der			
	Faktorisierung ertönt BEEP;			
	die Ergebnisse können mit			
	R/S oder XEQ 10 beliebig			
	oft abgerufen werden (Allerdings			
	auch bei gesetztem Flag 00 nur			
	in der gepackten Form von 2))			
•	Ist der Printer angeschlossen			
	und eingeschaltet, so werden			
	die Ergebnisse gemäß 2) oder			
	3) ausgedruckt.			
*	Wird das Programm ohne			
	angeschlossenen Printer eingetastet,			
	so wird Zeile 024 (siehe Listing			
	Seite 5) zu XEQ "PRA" (d.h. das			
	Programm wird um 3 Bytes			
	länger; die Funktion ändert sich			
	nicht).			
	Es darf dann im gesamten			
	Programmspeicher kein			
	globales Label "PRA" existieren.			

DESCRIPTION DU PROGRAMME I
DESCRIZIONE DEL PROGRAMMA I

Application, Equations, Variables
Anwendung, Gleichungen, Veranderliche
Application, Equations, Variables
Applicazioni, Equazioni, Variabili

Mit dem Programm wird für eine natürliche Zahl n $(n = \prod p^e)$ die kanonische Zerlegung in ihre Primfaktoren ermittelt.

Dabei wird im ersten Programmteil (E) unter Ausnutzung des Kongruenzverhaltens modulo 30 der Testteiler bis ca. $0{,}5 \cdot \sqrt{n}$ die Zahl faktorisiert.
Der dann verbleibende Restfaktor, der nur noch aus höchsten zwei Faktoren bestehen kann, wird über den Ansatz $n = (x-y)(x$ nach der Fermat'schen Methode (F) weiter untersucht, wobei für die Inkrementierung der x-Werte deren Kongruenzverhal modulo 144 ausgenutzt wird.

Literatur: D.E. Knuth, The Art of Computer Programming, Vol II, 4.5

Operating limits and Warnings
Grenzen und Einschränkungen
Limites et restrictions
Limiti operativi e avvertenze

Nach Verwendung des Programms als Unterprogramm müssen die Sonderfunktionsflags 12, 21, 25 und 29 und das Anzeigeformat ggf. neu gesetzt werden.

Die eingegebene Zahl n muß im Bereich $2 \leq n \leq 10^{10}$ liegen; falls sie nicht ganzzahlig ist, wird gerundet. Zu kleine oder zu große Eingaben werden vom Programm nicht abgefragt.

Example
Beispiel
Exemple
Esempio

Die Rechenzeit ist nicht nur vom Wert des größten Primfaktors abhängig, sondern auch von verschiedenen anderen Umständen.

Generell sind die Rechenzeiten für Zahlen n mit $n \bmod 3 = 2$ etwas länger als für Zahlen n mit $n \bmod 3 = 1$.

Die längste Rechenzeit ergibt sich für $n = 10^{10} - 71$; die beiden noch darüber liegenden Primzahlen unter 10^{10} werden schneller als solche erkannt, da sie kongruent 1 modulo 3 sind.

Die schnellsten Rechenzeiten ergeben sich bei nicht angeschlossenem Printer; bei ausgeschaltetem aber angeschlossenem Printer sind sie unwesentlich länger. Wesentlich länger sind sie bei eingeschaltetem Printer.

Es ist möglich, nach beendigter Faktorisierung den Printer anzuschließen, bzw. einzuschalten und dann mit R/S (bzw. XEQ 10) einen Ausdruck zu bekommen.

```
220
=2↑2*5*11

284
=2↑2*71

34765731
=3↑2*7*11*13
*17*227

36939357
=3↑2*7*13*23
*37*53

2172649216
=2↑8*257
*33023

2181168896
=2↑8*8520191
```

```
7,000000000 ENTER↑
7,999999999     Y↑X
5764900,989     ***
          XEQ "PRM"
5764801
=7↑8

9970592519
=2143*2153
*2161

          2 ENTER↑
         17 ENTER↑
          1       -
                Y↑X
          2 ENTER↑
         17     Y↑X
          1       -
                  *
8589869056
=2↑16*131071
```

```
        2,00 ENTER↑
        27,00   Y↑X
        111,40    -
   134217616,6  ***

134217617
IS PRIME

               SF 00
5907942612
=2↑2
*3↑2
*11↑2
*13
*17↑2
*19↑2

                 RUN
5907942612
=2↑2*3↑2
*11↑2*13
*17↑2*19↑2
```

PROGRAM LISTING

PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

Keystrokes / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commenti
01◆LBL "PRM" SF 05 SF 12 CF 29 FIX 0 PMD STO 00 2 STO 02 4 STO 04 + STO 06 SIGN 11 STO 08 + STO 09 CLA ASTO 11 ASTO 12 ARCL 00	Allgemeine Initialisierung
SF 21 SF 25 PRA CF 21 AVIEW	Test, ob Printer eingeschaltet
RCL 00 ENTER↑ ENTER↑ STO 07 RCL 02 MOD X=0? XEQ IND 04 CLX 3 MOD X=0? XEQ IND 04 CLX 5 MOD X=0? XEQ IND 04 CLX 7 MOD X=0? XEQ IND 04 XEQ 08	Vorab-Test der Faktoren p=2, 3,5 und 7 im Algor. E
52◆LBL 08 CLX RCL 04 LASTX + MOD X=0? XEQ IND 04 CLX RCL 02 LASTX + MOD X=0? XEQ IND 04 CLX RCL 04 LASTX + MOD X=0? XEQ IND 04 CLX RCL 02 LASTX + MOD X=0? XEQ IND 04 CLX RCL 04 LASTX + MOD X=0? XEQ IND 04 CLX RCL 06 LASTX + MOD X=0? XEQ IND 04 CLX RCL 02 LASTX + MOD X=0? XEQ IND 04 CLX RCL 06 LASTX + MOD X=0? XEQ IND 04 DSE 10 GTO 08	Testschleife für Algor. E (Kongruenz-verh. mod 30) (Schleifenindex Reg 10 wurde spätestens in Zeile 051 berechnet)
CLX LASTX X↑2 X>Y? GTO 97	Test, ob Faktorisierung abgeschlossen
ST+ Y CLX LASTX ST+ X / STO 10	Schleifenzähler für Algor. F (Rest Zeile 253)

Key pressed / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commenti
CF 06 CF 07 RCL 07 3 MOD GTO IND X	
128◆LBL 02 RCL 07 4 MOD 3 X=Y? GTO 03 18 STO 01 6 SF 06 GTO 02	
140◆LBL 03 12 STO 01 SF 07 GTO 02	
145◆LBL 01 RCL 07 12 MOD ST+ X RCL 07 9 MOD + XEQ IND X GTO 02	
156◆LBL 03 16 STO 01 2 RTN	
161◆LBL 06 18 STO 01 14 RTN 166◆LBL 09 8 STO 01 46 RTN 171◆LBL 15 20 STO 01 16 RTN	Berechnung der Inkremente für Algorithm. F (Kongruenz-verh. mod 144)
176◆LBL 18 4 STO 01 32 RTN	
181◆LBL 21 28 STO 01 9 RTN	
186◆LBL 02 STO 02 RCL 05 RCL 01 STO 03 ST+ X - RCL 02 - STO 04 RCL 07 16 MOD GTO IND X	
200◆LBL 03 201◆LBL 11 CF 07 RCL 01 X<> 02 STO 01 RCL 03 X<> 04 STO 03	
209◆LBL 01 210◆LBL 13 FS?C 06 GTO 11 RCL 02 X<> 04 STO 02	
216◆LBL 05 217◆LBL 07 218◆LBL 09 219◆LBL 15	

PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

Line / Zeile / Ligne / Linea	Keystrokes / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commenti
01	RCL 07 SQRT RCL X	Initialisierung
	RCL 05 MOD - STO 06	von x^2-n
	X↑2 RCL 07 -	
05	RCL 04 2 / FS? 07	Anfangsinkrem.
	CLX	für Algor. F
	XEQ 07 X=0? GTO 01	
10	RDN RCL 01 XEQ 07	Vorab-Test
	X=0? GTO 01 RDN	der ersten
	RCL 02 XEQ 07 X=0?	x-Werte
	GTO 01 RDN RCL 03	
	XEQ 07 X=0? GTO 01	
15		
	CLX RCL 10 RCL 06 -	
	RCL 05 / INT STO 10	Schleifenzähler
	ISG 10	für Algor. F
20		
	262•LBL 12	
	CLX RCL 04 X↑2 RCL 06	
	LASTX ST+ 06 ST+ X +	
25	+ + ENTER↑ SQRT FRC	
	X=0? GTO 01 CLX	
	RCL 01 X↑2 RCL 06	
	LASTX ST+ 06 ST+ X +	
	+ + ENTER↑ SQRT FRC	Testschleife
	X=0? GTO 01 CLX	für Algor. F
30	RCL 02 X↑2 RCL 06	
	LASTX ST+ 06 ST+ X +	(Kongruenz-
	+ + ENTER↑ SQRT FRC	verh. mod 144)
	X=0? GTO 01 CLX	
	RCL 03 X↑2 RCL 06	
35	LASTX ST+ 06 ST+ X +	
	+ + ENTER↑ SQRT FRC	
	X=0? GTO 01 DSE 10	
	GTO 12 GTO 97	
40		
	326•LBL 07	Test-U-Pgm.
	X↑2 RCL 06 LASTX	für mögliche
	ST+ 06 ST+ X + + +	negative
	ENTER↑ X<0? RTN SQRT	x^2-n Werte
45	FRC RTN	
	341•LBL 01	Faktor wurde
	RCL 06 LASTX - FRC	mit Algor. F
50	XEQ 04	gefunden

Line / Zeile / Ligne / Linea	Key pressed / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commenti
51	347•LBL 97	
	RCL 07 ENTER↑ SIGN	Faktorisierung
	STO 07 X<>Y X≠Y?	ist
	XEQ 01 BEEP XEQ 02	abgeschlossen
55	SF 21 GTO 02	
	359•LBL 10	
	SF 12 SF 21 CLA	
	ARCL 09 AVIEW RCL 09	
60	1 E3 / 11 +	Anzeige-
		Routine
	370•LBL 14	
	CLA ARCL IND X ISG X	RTN in
	ARCL IND X AVIEW ISG X	Zeile 384
65	GTO 14	ist RTN des
		Gesamtpro-
	378•LBL 02	gramms
	ADV CLX CLD CF 12	
	CF 21 RTN GTO 10	
70		
	386•LBL 04	Faktor gefunden
	RCL 07 LASTX MOD X≠0?	(Reduktion von
	GTO 01 CLX LASTX	n, Hochzählen
75	ST/ 07 SIGN ST+ Y RDN	des Exponenten)
	GTO 04	
	399•LBL 01	
	"*" FS?C 05 "="	Herstellung
80	RCL 00 LASTX ARCL X	der
	X=Y? "IS PRIME" SIGN	zusätzlichen
	R↑ X<=Y? GTO 01 "⊢↑"	Anzeige
	ARCL X	
	•	
85		
	414•LBL 01	
	FS? 25 SF 21 FS? 00	
	AVIEW CF 21 ASTO Z	
	ASHF ASTO T CLA	"Anhängen"
90	ARCL IND 08 ARCL IND 09	des neuen
	ARCL Z ARCL T ASTO 03	Anzeigeteils
	ASHF ASTO 01 ASHF	
	ASTO Y CLA ASTO X	
	X≠Y? GTO 01 RCL 03	
95	STO IND 08 RCL 01	
	GTO 13	
	441•LBL 01	
	XEQ 03 R↑ R↑ 2	neue Anzeige
100	ST+ 08 ST+ 09 RDN	zu lang
	STO IND 08 X<>Y	

PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

Line / Zeile / Ligne / Linea	Keystrokes / Tastenfolge / Touches / Tasti	Comments / Kommentar / Commentaires / Commento	Line / Zeile / Ligne / Linea	Key pressed / Tastenfolge / Touches / Tasti
01	451◆LBL 13		51	
	STO IND 09 ARCL IND 08	neue Anzeige		
	ARCL IND 09 FC? 00	paßt ins		
	AVIEW	Anzeigeregister		
05			55	
	457◆LBL 00			
	LASTX STO 01 RCL 07			
10	SQRT X<>Y - 30 /		60	
	RCL 07 3 MOD			
	XEQ IND X R↑ * INT	Berechnung		
	STO 10 RCL 07 RCL 07	des neuen		
	RCL 07 RCL 01 ABS RTN	Schleifenindex		
15		für Algor. E	65	
	480◆LBL 00	und der		
	481◆LBL 01			
	72 STO 05 ,48 RTN	Periodenlänge		
		in Algor. F		
	486◆LBL 02			
20	48 STO 05 ,557 RTN		70	
	491◆LBL 03	U-Pgm, das		
25	FC? 00 FC? 25 RTN CLA	bei eingeschalte-	75	
	SF 21 ARCL IND 08	tem Printer		
	ARCL IND 09 AVIEW CLA	eine fertige		
	CF 21 END	Anzeigezeile		
		druckt		
30			80	
35			85	
40			90	
45			95	
50			100	

REGISTERS, STATUS, FLAGS

REGISTERBELEGUNG, FLAGS, BETRIEBSARTEN
REGISTRES, INDICATEURS, MODES OPERATOIRE
REGISTRI, MODI OPERATIVI, FLAGS

Registers / Datenspeicher / Registres de données / Registri

Reg.	Inhalt	Reg.	Inhalt
00	n_0 Input (perm)	50	
01	Inkrement in F Anzeigehilfssp. Testfaktor p	55	
02	Inkrement in E " " F		
03	Inkrement in F Anzeigehilfssp.	60	
04	Inkrement in E " " F		
05	Periodenlänge in F	65	
06	Inkrement in E x in F	70	
07	n (reduziert)		
08	Index		
09	Index	75	
10	Zählvariable		
11–16	Anzeigeregister	80	
		85	
40		90	
45		95	
		99	

Status / Kenngrößen / Modes opératoires / Modi operativi

Size 017 Pgm 099

						User Mode	
Eng	☐	Fix	☒ 0	Sci	☐	On	☒
Deg	☒	Rad	☐	Grd	☐	Off	☐

Purpose / Bedeutung / Signification / Scopo — **Flags**

Flag	Bedeutung (SET)	SET	CLEAR	Bedeutung (CLEAR)
00	Einzelanzeige	X	X	gepackte Anzeige
01				
02				
03				
04				
05	kein Faktor gef.	X	X	erster Faktor gef.
06	n mod 4 = 1	X		
07	n mod 4 = 3	X		
08				
09				
10				
11	Auto execute			
12		X		
13				
14				
15				
16				
17				
18				
19				
20				
21	Printer Enable	X	X	
22	Number Input			
23	Alpha Input			
24	Range Ignore			
25	Error Ignore — Printer "on"	X	X	Printer "off" oder nicht angeschlossen
26	Audio Enable			
27	User Mode			
28	Decimal Point			
29	Digit Grouping		X	

Assignments / Tastenbelegung / Assignations / Assegnamenti

Function / Funktion / Fonction / Funzione	Key / Taste / Touche / Tasto	Function / Funktion / Fonction / Funzione	Key / Taste / Touche / Tasto
PRM	14		

Ein einfaches Problem - eine ebenso einfache Lösung ?!

Das Problem ist auf den ersten Blick tatsächlich recht simpel:

Drei Werte sollen sortiert werden.

Das gängige Sort-Programm für den 41er dafür einzusetzen, halte ich für eine zu einfache Lösung. Also muss ein neues Programmchen her! So gedacht, so getan. Nach etwa einer halben Stunde hatte ich's dann auch.

Wie lange habt Ihr wohl dazu ? Ich gebe zu, dass ich das Problem recht lange gewälzt habe. Nun lief es jedenfalls. Doch das genügte mir noch nicht. Eine BASIC-Version musste her. Doch da merkte ich, dass es so 'einfach', wie mit dem 41er gings da nicht mehr. Zwar braucht das Programmchen nur zwei (Verarbeitungs-)Zeilen, aber ich brauchte auch ein Hilfsfeld. Die PL/1-Version war dann schon etwas einfacher (zwar ein wenig viel Schreibarbeit), obwohl ich im ersten Test einen Dump produzierte (was soviel heisst, wie, dass der Rechner - eine IBM 370/148 - ins 'Kraut' gefahren ist).

Beiliegend jedenfalls die 41er und die PL/1 Versionen. Bevor Ihr die 41er-Version ausprobiert, versucht doch, dieses Problemchen selber zu lösen! Eine gute Uebung, nicht nur für Anfänger!

Werte in R_{01} bis R_{03} gespeichert

```
LBL "SORT3"
RCL 02
RCL 01
x ≤ y?
GTO 00
x<>02
STO 01
LBL 00
RCL 03
x > y?
GTO 01
x<>y
x<>02
x<>03
x<>y
STO 01
LBL 01
18 .END.
```

Kurze BASIC-Version

```
10 REM   TAB=SORT=3=ELEMENT
20 INPUT T1,T2,T3
30 IF T1  T2 THEN A=T2:T2=T1:T1=A
40 IF T1  T3 THEN A=T3:T3=T2:T2=T1:T1=A
50 PRINT"SORT: 1.";T1;"2.";T2;"3.";T3
60 END
```

Viel Spass beim ausprobieren!

(Das PL/1 Programm ist eine ganze A4 Seite lang. ich drucke es daher lieber nicht mit ab...Oliv.

Vielleicht findet jemand eine noch kürzere Version!? Schickt sie an Oliver zur Veröffentlichung.

M I T T E I L U N G der HP-41C-Anwender-Programm-Bibliothek, Dornach

---

Ich möchte allen recht herzlich für ihre Unterstüzung danken. Leider bin ich - durch persönliche Probleme - gezwungen, den Betrieb vorläufig einzustellen. Ich werde mich jedoch bemühen, das ganze bis Herbst '81 wieder in Gang zu bringen. Bis dahin werde ich ausserdem in der Lage sein, die - etwas voreilig angekündigten - Lösungsbücher bereitzustellen. Ich wünsche allen Clubmitgliedern einen schönen und nicht zu heissen Sommer.

Happy Programming

G u i d o Studer

Für Programme bin ich natürlich auch während dieser Zeit dankbar!
HP-Anwender-Prgm.-Bibliothek, Postfach 18, 4143 Dornach (Schweiz)

Für die Abdruckerlaubnis
sei HP vielmals gedankt!

HP 82104A Magnetkartenleser Kurzanleitung

Wenn der Magnetkartenleser HP 82104A an den HP-41C angeschlossen ist, werden die nachfolgend beschriebenen Funktionen im System wirksam. Diese Funktionen und Programme, die diese Funktionen beinhalten, können nur ausgeführt werden, solange der Kartenleser an den Rechner angeschlossen ist. Sie können ein vollständiges Verzeichnis dieser Funktionen mit [CATALOG] 2 auflisten.

Funktionsverzeichnis

[MRG] Hängt eingelesene Programme im Speicher an. Ersetzt alle Programmzeilen ab der augenblicklichen Position im Speicher durch das Magnetkarten-Programm. Der HP-41C muß auf das letzte Programm positioniert werden oder [MRG] Bestandteil des letzten Programms sein.

[RDTA] Einlesen von Datenkarten. Die aufgezeichneten Karten werden ab R_{00} in den Speicher übernommen.

[RDTAX] Einlesen von Datenkarten, gesteuert durch eine Zahl in X. Format von X ist bbb.eee, wobei bbb die Anfangs- und eee die Endadresse ist.

[RSUB] Einlesen eines Unterprogramms. Ersetzt das letzte Programm im Speicher durch das Magnetkarten-Programm. Wenn [RSUB] Bestandteil des letzten Programms ist oder der HP-41C auf das letzte Programm positioniert ist, ([RSUB] über das Tastenfeld ausgeführt), wird das Magnetkarten-Programm hinter dem letzten Programm im Speicher angehängt.

[VER] Verifiziert (überprüft) Magnetkartenspur (nicht programmierbar). Zeigt TYPEtTRnn an; dabei ist t wahlweise: P (Programm), D (Daten), S (Status), A (Speicherauszug), 7P (HP-67/HP-97 Programm), 7D (HP-67/HP-97 Daten); nn ist die Nummer der Spur. Fehlerhafte Daten verursachen CHECKSUMERR. Unbeschriebene oder unlesbare Karten verursachen CARD ERR.

[WALL] Schreiben von Speicherauszugskarten (nicht programmierbar). Zeichnet alle im HP-41C enthaltenen Informationen auf.

[WDTA] Schreiben von Datenkarten. Zeichnet die Inhalte sämtlicher definierten Speicherregister ab R_{00} auf.

[WDTAX] Schreiben einer Datenkarte, gesteuert durch eine Zahl in X. Format von X ist bbb.eee, wobei bbb Anfangs- und eee Endadresse ist.

[WPRV] Schreibt „Privat"-geschützte Programmkarte (programmierbar). Zeichnet augenblickliches Programm auf.

[WSTS] Schreiben von Statuskarten. Zeichnet den Status der Flags 0 bis 43 auf, Position der Statistik-Register, Zahl der Datenspeicherregister, Tastenbelegungen der Standard- und Erweiterungs-Funktionen sowie Inhalte von X, Y, Z, T, LAST X und ALPHA-Register.

[7CLREG] Löscht R_{00} bis R_{09}, R_{20} bis R_{25}. Entspricht HP-67/HP-97 [CL REG]

[7DSP0] bis [7DSP9] Wählt 0 bis 9 anzuzeigende Nachkommastellen. Entspricht HP-67/HP-97 [DSP] 0 bis 9.

[7DSPI] Wählt 0 bis 9 Nachkommastellen bei indirekter Steuerung durch R_{25}. Entspricht HP-67/HP-97 [DSP] [(i)].

[7DSZ] Dekrement und Sprung wenn Null. Entspricht HP-67 [DSZ] und HP-97 [DSZ] [I].

[7DSZI] Dekrement und Sprung wenn Null bei indirekter Steuerung durch R_{25}. Entspricht HP-67 [DSZ (i)] und HP-97 [DSZ] [(i)].

[7ENG] Technisches Anzeigeformat. Ändert nicht die Zahl der angezeigten Nachkommastellen. Entspricht HP-67/HP-97 [ENG].

[7FIX] Festkomma-Format. Ändert nicht die Zahl der angezeigten Nachkommastellen. Entspricht HP-67/HP-97 [FIX].

[7GSBI] Sprung zum Unterprogramm bei indirekter Steuerung durch R_{25}. Entspricht HP-67/HP-97 [GSB] [(i)]. Negative Werte führen zu NONEXISTENT

[7GTOI] Sprung zu einer Marke bei indrekter Steuerung durch R_{25}. Entspricht HP-67/HP-97 [GTO] [(i)]. Negative Werte führen zu NONEXISTENT

[7ISZ] Inkrement und Sprung wenn Null. Entspricht HP-67 [ISZ] und HP-97 [ISZ] [I].

[7ISZI] Inkrement und Sprung wenn Null bei indirekter Steuerung durch R_{25}. Entspricht HP-67 [ISZ (i)] und HP-97 [ISZ] [(i)].

[7P<>S] Austausch der Inhalte von Primär- und Sekundär-Registern. Vertauscht R_{00} bis R_{09} gegen R_{10} bis R_{19}. Entspricht HP-67/HP-97 [P⇄S].

[7PRREG] Listet bei angeschlossenem Drucker die Inhalte von R_{00} bis R_{09}, R_{20} bis R_{25} auf. Bei nicht angeschlossenem Drucker werden zuerst die Registernummer und dann der Inhalt angezeigt. Entspricht HP-67/HP-97 [REG].

[7PRSTK] Druckt bei angeschlossenem Drucker T, Z, Y, X. Anzeige, wenn kein Drucker angeschlossen. Entspricht HP-67 [STK] und HP-97 [STACK].

[7PRTX] Druckt bei angeschlossenem Drucker X-Registerinhalt. Zeigt X an, wenn kein Drucker angeschlossen. Entspricht HP-67 [-x-] und HP-97 [PRINTX].

[7RCLΣ] Ruft Inhalte der Statistikregister zurück. Entspricht HP-67/HP-97 [RCL] [Σ+].

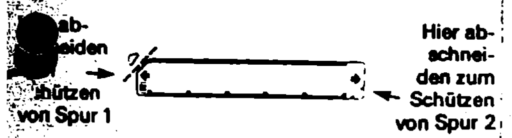
[7SCI] Wissenschaftliches Anzeigeformat. Ändert nicht die Zahl der angezeigten Nachkommastellen. Entspricht HP-67/HP-97 [SCI].

Magnetkarten - Arten

Spur 1 (16 Register) — Spur 2 (16 Register)

Programmkarten enthalten Programmanweisungen. Datenkarten enthalten Daten aus den Speicherregistern. Statuskarten enthalten Status-Informationen (siehe [WSTS]). Speicherauszugskarten enthalten sämtliche Informationen aus dem HP-41C.

Geschützte Magnetkarten



Wenn der Eckenabschnitt zu einer Spur entfernt ist, können keine weiteren Informationen auf dieser Spur aufgezeichnet werden. Beschnittene Karten können aber normal eingelesen werden. Zum bewußten Überschreiben einer geschützten Karte ist das Überschreibungs-Flag (Flag 14) zu setzen.

Aufzeichnen von Programmen auf Magnetkarten

Positionieren Sie den HP-41C im PRGM-Modus auf das gewünschte Programm und führen Sie eine Karte ein. Der HP-41C fordert die Karten mit

an, wobei kk die als nächstes zu schreibende Spur und nn die Gesamtzahl der erforderlichen Spuren ist. Zum Schreiben „Privat"-geschützter Programme führen Sie [WPRV] aus. Diese Programme können nicht angezeigt, geändert oder erneut aufgezeichnet werden.

Einlesen von Programmkarten

Programmkarten können auf drei verschiedene Weisen eingelesen werden: 1. Schieben Sie die Karte ein, wobei sich der HP-41C nicht im PRGM-Modus befindet; 2. führen Sie [RSUB] aus; 3. führen Sie [MRG] aus. Der HP-41C fordert die Karten mit

an, wobei kk die am niedrigsten nummerierte Spur ist, die noch nicht gelesen wurde, und nn die Gesamtzahl der zu lesenden Spuren. Magnetkarten können in beliebiger Reihenfolge eingelesen werden. Lesevorgang kann jederzeit mit [R/S] oder [←] abgebrochen werden.

Automatische Programmausführ.

Setzen Sie Flag 11 vor dem Schreiben der Programmkarte. Der Rechner beginnt dann die Ausführung des Programms bei jedem Einlesen mit Zeile 0; dabei ertönt ein Piepton.

Aufzeichnen von Daten auf Kart

Zwei programmierbare Funktionen werden zum Aufzeichnen der Speicherregisterinhalte auf Magnetkarten verwendet: [WDTA] und [WDTAX]. Der HP-41C fordert die Karten mit an.

Einlesen von Daten von Karte

Drei Möglichkeiten zum Einlesen von Datenkarten: 1. Führen Sie die Karte ein, wobei sich der HP-41C nicht im PRGM-Modus befindet (auf der Karte gespeicherte Daten werden ab R_{00} in den Speicher geladen); 2. führen Sie [RDTA] aus; 3. führen Sie [RDTAX] aus. Der HP-41C fordert die Karten mit RDY kkOFnn an, wobei kk die am niedrigsten nummerierte Spur ist, die noch nicht gelesen wurde, und nn die Gesamtzahl der zu lesenden Spuren. Karten können in beliebiger Reihenfolge eingelesen werden. Der Lesevorgang läßt sich jederzeit mit [R/S] oder [←] beenden.

Erstellen von Statuskarten

Führen Sie [WSTS] aus, wobei sich der HP-41C nicht im PRGM-Modus befindet. Der HP-41C fordert die Karten mit RDYkkOFnn an.

Einlesen von Statuskarten

Überzeugen Sie sich, daß der HP-41C nicht in den PRGM-Modus geschaltet ist und führen Sie Spur 1 zuerst ein. Der HP-41C fordert die Karten mit RDY kkOFnn an.

Speicherauszugskarten

Zum Schreiben, führen Sie [WALL] aus. Der HP-41C fordert die Karten mit RDY kkOFnn an. Vorzeitiges Abbrechen des Schreibvorgangs führt zu einem unvollständigen [WALL] Kartensatz, der nicht eingelesen werden kann.

Überzeugen Sie sich zum Einlesen des Kartensatzes davon, daß der HP-41C nicht in den PRGM-Modus geschaltet ist und führen Sie den vollständigen [WALL] Kartensatz in beliebiger Reihenfolge ein. Unterbrechung des Lesevorgangs führt zu

MEMORY LOST

XROM Funktionen

Kartenleserfunktionen in Programmen werden als **XROM**-Nummern angezeigt, während der Kartenleser nicht an den HP-41C angeschlossen ist. Einzelheiten zu **XROM**-Nummern können Sie im HP 82104A Magnetkartenleser Bedienungshandbuch nachlesen.

HP-67/97 Kompatibilität

HP-67/HP-97 Magnetkartenprogramme werden zur Verwendung mit dem HP-41C übersetzt (siehe Kompatibilitätsfunktionen im Funktionsverzeichnis). Zur Ausführung übersetzter Programme ist der HP-41C in den USER-Modus zu schalten und dann die Taste für die entsprechende Marke zu drücken (A bis E, ■ a bis ■ e), wie es beim HP-67/HP-97 erforderlich ist. Die Tasten der oberen Tastenfeldzeile dürfen dabei nicht mit anderen Funktionen bzw. Programmnamen belegt (ASM) sein.

HP-41c Register	HP-67/97 Reg
R_{00} bis R_{09}	Primär-Register
R_{10} bis R_{19}	Sekundär-Register
R_{20} bis R_{24}	A bis E
R_{25}	I

HEWLETT PACKARD

82104-90012
German- 8.79-Ca
Printed in U.S.A.

```
01♦LBL "KEGELN"
FIX 0  CF 21  SF 27
CF 29  CLRG  2  STO 00

09♦LBL B
1  ST+ 00  RCL 00  AON
"NAME ?"  PROMPT  AOFF
ASTO IND X  1  +  STO 01
1  STO IND 01  2  ST+ 00

25♦LBL 06
CLA  RCL 01  1  -
ARCL IND X  "├ *"
ARCL IND 01  "├*"  AVIEW
CF 22

36♦LBL b
PSE  FC? 22  GTO b  9
X<Y?  GTO 06  RDN
TONE 7  XEQ 00
VIEW IND Y  RDN  STO 00
1  ST+ IND 01  PSE
GTO 06

53♦LBL E
1  ST- IND 01  XEQ 01
RCL IND 00  RCL Z  /
INT  10  /  FRC  10  *
CHS  XEQ 00  GTO 06

69♦LBL 00
XEQ 01  *  ST+ IND Y
RTN

74♦LBL 01
RCL IND 01  9  +  10  /
INT  RCL 01  +
RCL IND 01  1  -  10  /
FRC  10  *  9  -  ABS
10↑X  RCL Z  RTN

97♦LBL 02
CLX  STO 02  1  ST+ 00
RTN

103♦LBL D
CF 00  SF 01  SF 02
SF 21  2  STO 00

110♦LBL 07
1  ST+ 00  RCL IND 00
SF 25  1  *  FC?C 25
GTO 08  RCL 02  9  X>Y?
SF 00  10  ST- 02  .009
STO 03  CLX  STO 01  CLA
RCL IND 00  X=0?  GTO 11
ARCL X  1 E9  /

136♦LBL c
XEQ 10  ISG 03  GTO c
"├ "  ARCL 01  "├ "
FS?C 01  XEQ 11  RCL 01
ST+ 04  ARCL 04  FS? 00
XEQ 09  FC? 00  70  -
ST+ 05  "├ "  ARCL 05
PRA  GTO 07

158♦LBL 08
ADV  SF 12  "** "
ARCL IND 00  "├ **"
AVIEW  1  ST+ 00  " "
RCL IND 00  1  -  ARCL X
"├ WURF"  AVIEW
RCL IND 00  STO 02
CF 12  ADV  CLX  FC? 02
STO 04  FC?C 02  STO 05
SF 01  GTO 07

185♦LBL 09
RCL 02  X=0?  GTO 11  9
+  7  *  RTN

194♦LBL 10
ENTER↑  INT  ST+ 01  RDN
FRC  10  *  RTN

203♦LBL 11
"├ "  0  STO 04  STO 05
END

 LBL'KEGELN
 END         356 BYTES
```

```
                XEQ "KEGELN"
NAME ?
OLIVER
OLIVER *1*
                 7000000000
OLIVER *2*
                 7700000000
OLIVER *3*
                 7770000000
OLIVER *4*
                 7777000000
OLIVER *5*
                 7777800000
OLIVER *6*
                 7777870000
OLIVER *7*
                 7777878000
OLIVER *8*
                 7777878800
OLIVER *9*
                 7777878870
OLIVER *10*
                 7777873877
OLIVER *11*
                 7000000000
OLIVER *12*
                 7700000000
OLIVER *13*
                 7760000000
OLIVER *14*
```

K E G E L N

Hier ein kleines Programmchen für alle Freunde des Kegelsportes: Ein vollautomatischer Kegelnotizkalender mit integrierten Druck-und Rechenroutinen.

Problemstellung:

Abspeichern einer unbekannten Anzahl einstelliger Zahlen. Danach Ausdruck in Zehnerblöcken, die jeweils addiert werden, außerdem Ausdruck von der Summe aller bisherigen Würfe und die Abweichung vom Durchschnitt 7.

Bedienung des Programms:

1) Eingeben oder Einlesen (Strichcodes werden abgedruckt! Magn.K.bei Walter)
2) Start: XEQ"KEGELN"
3) Namen des ersten (bzw.nächsten) Keglers eingeben (bis 6 Zeichen)
4) Nacheinander die Würfe dem Rechners eingeben: Rechner zeigt Wurfnummer an, einfach Taste drücken (Ø bis 9). Es muß einmal piepen (Eingabekontrolle). Zur Kontrolle wird immer der derzeitige Zehnerblock angezeigt. Wer sich bei der Eingabe vertut, drücke einfach E !!
5) Erschöpft möge der Besitzer von der Bahn kriechen, den Printer anschließen und dann die Taste D drücken:Die komplette Tabelle wird ausgedruckt !! ...und alle Kegelschwestern, -brüder beginnen zu staunen !
6) Der nächste Kegler kann natürlich auch erst einmal kegeln, bevor alle Kegelpläne zusammen ausgegeben werden ! Einfach Taste B drücken und weiter bei 3)

Allgemeines:

Mit dem 41cv lassen sich viele tausend Würfe auf einmal abspeichern (= mehrere Mannschaften), zudem Namen usw.; auch der HP-41c ohne Memory reicht für einen Kegler schon aus. DiesProgramm zeigt sehr deutlich, wieviel Datenspeicher man zur Verfügung hat, wenn manDigitweise abspeichert! Und es hat sich bereits bestens bewährt !!!!! Nur: Der Rechner ist die ganze Zeit im RUN-Modus und säuft entsprechend viel: Aufpassen !! Geschrieben 198o und bereits 8 Monate anstandslos im Test !   Oliver Rietschel (1).

BEISPIEL:

** OLIVER **
100 WURF

7777878877	73	73	3
7767778775	68	141	1
7676679778	70	211	1
7778787776	71	282	2
7777797757	70	352	2
7878776776	70	422	2
7877677579	70	492	2
7787757878	71	563	3
7577787697	70	633	3
5978777887	72	705	5

** KURT **
100 WURF

7787778767	71	71	1
7787778767	71	142	2
9798877787	77	219	9
7888857767	71	290	10
7677787787	71	361	11
8768778877	73	434	14
7787778767	71	505	15
5887788878	74	579	19
7777878877	73	652	22
7778877677	71	723	23

Liebe Mitglieder !

Die letzte prisma-Ausgabe (Juni) trieb den Arbeitsaufwand ins nahezu Unüberwindbare: Allein der Versand dauerte über einhundert Arbeitsstunden und zog sich über zwei Wochen hin. Wenngleich zu Pfingsten alle Mitglieder das Info in Händen hatten, brach der Versand an neue Mitglieder zeitweise zusammen.
Aus diesen Gründen werden wir uns nach anderen Versandarten umsehen und diese Ausgabe zunächst quantitativ ein wenig einschränken. Dennoch wünsche ich viel Spaß, denn die Qualität wird keineswegs leiden ..... Alles Gute Euer Oliver

---

- Mitglieder des PPC haben es bereits vor einigen Wochen erfahren: Page-Switching heißt eine Methode, mit der man bis zu acht Quad-Memorys an den HP-41c anschließen kann. Diese Module können dann als Permanentspeicher dienen, wobei jeweils ein Quadram im direkten Zugriff ansprechbar ist. Die Module können Hard- und Softwaremäßig geschaltet werden. Aber Vorsicht: Wer nach der Anleitung im PPC-Journal vorgeht, kann sehr leicht auf die Nase fallen: Die PPC-Schaltung ist falsch ! Da nach einer Umfrage nur extrem wenig Clubmitglieder in der Lage wären, den Umbau zu bewältigen, seien alle aufgerufen, die neu eingeführte Clubbörse zu beachten ! Dort findet sich ein Angebot von Hans-Günter Lütke Uphues.

- Magnetkartenfälschungen: Neuerdings tauchen massiv Kartenfälschungen auf, die den HP-Magnetkarten sehr ähnlich sind und optisch praktisch nicht zu unterscheiden sind Qualitativ sind sie jedoch wesentlich schlechter, so daß eine "vernünftige" Anwendung nicht möglich ist. Wer solche Karten irgendwo erstanden hat, sollte unbedingt versuchen, sie umzutauschen. Die Karten wurden nicht von HP produziert.

- Wir haben ein neues interessantes Angebot von einem Computerhändler bekommen, so daß die Möglichkeit eines noch günstigeren Clubeinkaufs in Aussicht steht. Im positiven Fall findet sich ein entsprechender Bericht im nächsten prisma !

- Vereinzelt lassen sich in den USA derart günstige Preise erzielen, daß selbst die deutschen Großhändler noch teurer sind. Wer also in die USA reist, möge sich bitte kurz einmal bei Ulli (3) melden. Bei dem Dollarkurs von 2.45 DM habe ich allerdings rege Bedenken ...

- Auf dem Konto von Ulli (3) ist eine Zahlung in Höhe von 168,—DM ohne Absender eingegangen. Sofern es kein großzügiges Geschenk eines anonymen Wohltäters ist, möge sich der jetzt 168,-- ärmere Mensch bitte bei Ulli melden.

- Eine kleine Clubstatistik: Seit Bestehen unseres Clubs wurden über Ulli fast 15 ooo Magnetkarten mit Taschen und nahezu 15oo Rollen Tehrmopapier bestellt. Ungefähr 1o ooo Magnetkarten und schätzungsweise 1 ooo Papierrollen wurden von Mitgliedern mit Bedarf von über 2 ooo Karten und/oder 3oo Rollen direkt bestellt. Davon wurden allein im ersten Halbjahr 1981 24 ooo Karten und alle 2 5oo Papierrollen bestellt. Das macht je Mitglied 5 Papierrollen und 48 Magnetkarten !

- Den letzten Info fehlten die Seiten ooo-81 und 127-81. Diese Seiten liegen diesem Info bei !

- In Kegeln 1 und FN 3 wurden beim Plotten der Strichcodes jeweils ein Bit vertauscht. Die Folgen sind katastrophal, so daß diese Seiten erneut geplottet werden. Können die Synthetic-Fans den RAM-Killer von FN 3 erklären ????

Lösungsbücher:

Die Nachfrage nach dem Lösungsbuch Nr. 1 hat deutlich nachgelassen. Der Preis bleibt nach wie vor bei 15,90 DM (Selbstkosten). Dagegen gingen viele Anregungen und Verbesserungsvorschläge ein, die bei den beiden folgenden Büchern - so hoffe ich - weitestmöglich berücksichtigt wurden. Aus organisatorischen Gründen muß allerdings folgende Konvention getroffen werden: LöBu 1 und zwei sind bei Oliver erhältlich, die anderen alle bei dem jeweiligen Autor. Angeboten werden die Lösungsbücher in der neuen "Clubbörse" vom jeweiligen Autor, womit sie unter Alleinregie des Autor vertrieben werden. Eine andere Möglichkeit besteht bei der derzeitigen Clubstruktur nicht.

Lösungsbuch 1: Spiele
Lösungsbuch 2: Kalenderrechnungen
Lösungsbuch 3: Intelligente Spiele

- Clubbörse: Auf den Wunsch einiger Mitglieder wollen wir einen Versuch starten, unseren Club noch attraktiver zu gestalten. Bisher wurden nur vereinzelt Gesuche oder Angebote von Clubmitgliedern veröffentlicht. Denen soll nun eine eigene Abteilung eingeräumt werden: Gelegenheitsanzeigen aller Art, sofern sie mit den Clubaktivitäten zusammenhängen, können von allen Mitgliedern kostenlos aufgegeben werden. Den Anfang machen wir gleich mit einer Lösungsbuchvorstellung und - vielerorts sehnsüchtig erwartet - einer Produktvorstellung: Kassetteninterface von Ulrich Jansen für den HP-41c(v) !! Außerdem Page Switching von Hans-Günter und vieles mehr !!!!!! Aber auch wer irgendetwas kaufen oder verkaufen will etwas sucht oder sonst irgendetwas will, möge an Oliver eine Anzeige mit Schreibmaschine geschrieben schicken.

- Irgendwo in dieser Ausgabe wird sich ein Bericht "Wissenswertes für Neulinge" finden, den auch manch ein "alter Hase" ruhig einmal lesen sollte....

- Das unser Club keineswegs nur aus den Infos besteht, wollen wir wieder einmal beweisen:

Übersicht der in der Clubbibliothek - Frank Altensen, Uhlandstr. 9, 6365 Rosbach 3 - derzeit enthaltenen Bücher

1) 1 x H.H. Gloistehn, "Programmierung von Taschenrechnern 1, SR-56"
2) 2 x H. Alt, "Angewandte Mathematik, Finanzmathematik, Statistik, Informatik für UPN-Rech
3) 1 x H. Schumny, "Taschenrechner + Microcomputer Jahrbuch 1981"
4) 4 x W.C. Wickes, "Synthetic Programming on the HP-41c" (engl.)
5) 3 x HP-Lösungsbücher "Games" (engl.)
6) 3 x HP-Lösungsbücher "Geometry" (engl.)
7) 2 x HP-Lösungsbücher "Calendars" (engl.)
8) 1 x HP-Lösungsbuch "Statistiks" (engl.)
9) 1 x HP-Lösungsbuch "Chemistry" (engl.)
10) 1 x HP-Lösungsbuch "Maschinenbau" (deutsch) - bestellt -
11) 1 x HP-Lösungsbuch "High level Mathematics" (deutsch) - bestellt -
12) 1 x EDV für Schulen Band 1
13) 1 x EDV für Schulen Band 2
14) 1 x TI-59-Specialty Pakettes "Lab Chemistry"

Alle Bücher können auch per Post entliehen werden: Wattierte Versandtasche, größer DINA4, und je Buch 1,40 Porto, max. 2,80 einsenden und Absender und gewünschtes Buch angeben. Keinesfalls die Mitgliedsnummer vergessen !

Nur zur Einsichtnahme sind in der Bibliothek vorhanden:

E1) 1 x HP-Games-Pac-Module (engl.)
E2) 1 x HP-Network-Analysis-Module (engl.)

W E R stiftet noch ein Büchlein ???

Die Barcode Seiten Kegeln und FN sind mit 188-81 bis 193-81 zu bezeichnen (Kegeln-1- = 188-81). Oliver

Nur HP-41c(v) ?? oder: Ein rein subjektiver Bericht über Personalcomputer

In versch. Gesprächen werde ich immer häufiger gefragt, was ich von diesem oder jenen Computer halte. Selbstverständlich kann ich dann immer nur meine persönliche Meinung äußern - die nach jahrelanger Praxis als Informatik-Assistent keineswegs unqualifiziert ist. Diese Anfragen nun nahmen in letzter Zeit immer mehr zu, so daß ich mich zu diesem Bericht bequeme.
Wie auch ich wird sich manch ein 41 Besitzer fragen, ob sich nicht die Anschaffung eines Personalcomputers lohnt. Doch beginnt das Erlebnis nur bei der Schreibmaschinentastatur - meistens leider eine amerikanische - und findet seinen Höhepunkt bei den Erfahrungen mit dem Betriebssystem des Rechners um in einer derben Enttäuschung oder grenzenloser Freude zu enden. Von mir nun wird immer die Antwort erwartet: DER Computer und kein anderer. Doch gäbe es diesen schönen Computer - ich hätte ihn längst.
Bisher habe ich auf größeren Rechnern (Wang,Cromemco,Siemens,Northstaru.ä.) Software hauptsächlich zu Lehrzwecken erstellt. Durchaus empfehlenswert ist darunter nur das Cromemco-System, dem aber alle neuen CPM-Systeme nur in wenigen Sachen nachstehen (Vektor, Altos, Adler, Alpha, CTM, IBM, HP, Dynabyte, IMS, Nixdorf,(Wang), u.a.).
Alle diese Systeme haben aber durchweg eines gemeinsam: Unter 15 ooo,--DM sind sie bestimmt nicht zu haben. Damit sind sie aber für den Hobby-Markt uninteressant, der sich deswegen auf Geräte wie TRS-8o, Apple, CBM, Sorcerer und Sinclair richtet.
Bei unserem HP-Besuch in der vergangenen Woche bekam ich nun nicht nur die Möglichkeit, den HP-85 kennenzulernen, sondern sah mir auch gleich den Apple einmal näher an. Dabei ging es mir weniger um die Hardware als um die Software, denn hardwaremäßig tuen sich bei diesen Hobbygeräten keine solch großen Unterschiede auf, daß sie für den Hobbyanwender ernstliche Kaufgründe wären: Fast alle wichtigen Dinge lassen sich nachrüsten, die Frage richtet sich vielmehr nach dem Preis.
Von dem Großrechner-Luxus wie Cromemco-Structured-Basic u.ä. verwöhnt setzte ich mich also an den lieben Apfelrechner und kann mein Erlebnis als schockierend bezeichnen. Die Apple-Sprachen empfand ich als extrem umständlich, unübersichtlich und verwirrend. Deutlich war zu erkennen, daß der Apple einer der ersten Personalcomputer war (ist) und dementsprechend veraltet..... . Leider steht es auch mit TRS-8o und CBM nicht anders, so daß ich meinen Blick auf den Sorcerer schweifen ließ und dort schon etwas angenehmere Sachen erblickte. Mit der S-1oo Box und einem CPM-DOS läßt sich praktisch jede vernünftige CPM-Sprache verwenden, womit der Sorcerer seinen Apple und sonstigen Konkurrenten kilometerweit überlegen ist - insbesondere aus Sicht des Nicht-Nur-Spielers. Also- warum empfehle ich nicht den Sorcerer ?
Gerne würde ich es tun, denn mit der Z-8o CPU (die ein Personalcomputer wenigstens haben sollte), vernünftiger Tastatur mit Groß- und Kleinschreibung (ganz im Gegensatz zum Apple), 64x24 Zeichen Bildschirm (Apple 4o x 16), CPM-DOS (wichtig) und sehr vielseitigem S-1oo Bus ist der Sorcerer allen seinen Konkurrenten weit überlegen. Doch der Haken liegt im CPM, das die Minifloppys (SS,SD) auf rund 25oo,--DM je Stück verteuert. Zur vernünftigen Anwendung braucht man zwei Floppys, S-1oo Box, gutes CPM-Betriebssystem (strukturiertes Basic) - und liegt kurzum auch über 1oooo,--.
Daher kann ich allen Hobbyprogrammierern, die mit dem Gedanken spielen, sich einen Rechner dieser Art zu kaufen, nur zuflüstern: Wartet noch ein bischen ! Ich tu es ja auch ....
Wer jetzt aber nicht so lange warten will, dem kann ich eine Empfehlung geben: Wird der Computer nicht für Textverarbeitung gebraucht, so ist der HP-85 in die nähere Auswahl zu ziehen. Gründe: Sein Betriebssystem ist allen Konkurrenten überlegen, deutlich überlegen. Bandlaufwerk und Thermodrucker reichen für den Hobbyisten völlig aus, die Tastatur ist vernünftig und überhaupt: Qualität wird- HP üblich - GROßgeschrieben.
Was ich bereits nach kurzem verspürte dürfte letztlich für jeden Hobbyisten entscheidend sein: nach einigem Einüben bringt das Programmieren Spaß - es ist wirklich eine Freude. Nur: Wer an Textverarbeitung denkt, sollte den 85 schnell vergessen, so daß man CHIP zitieren darf: Fast ein Rolls-Royce ! Das ist auch der einzige Grund, warum ich noch keinen 85 mein eigen nenne: Der Rechner (er hat auch die Strichcodes in dieser Ausgabe geplottet) ist ausgezeichnet - der Club aber erfordert viel Textverarbeitung - das einzige Bauteil vom Volkswagen am 85.
Also liebe Clubmitglieder - nervt mich nicht mehr mit Anfragen dieser Art - meine Meinung kennt ihr: Abwarten bringt nicht nur Freude, sondern auch Geldersparnis. Zunächst sollte der 41c völlig ausreichen. VIEL SPAß !

Oliver

Wissenswertes für Neulinge

Unser Club ist wieder einmal um hundert Mitglieder gewachsen - und die Organisation wird entsprechend schwieriger. Daher wendet sich dieser Artikel an alle Mitglieder, auch die "alten Hasen":
Alle den Club betreffenden Schreiben können nur beantwortet werden, wenn neben dem vollständigen Absender deutlich die Mitgliedsnummer angegeben ist ! Dies gilt für alle Schreiben, auch z.B. für Bestellungen bei Fi. Hansen in Bonn ! Wer seine Mitgliedsnummer vergessen hat, findet sie grundsätzlich rechts oben auf dem Adressaufkleber der ihm zugesandten Infos. Auch Überweisungen (Beitrag etc.) sind immer mit der Mitgliedsnummer zu versehen ! Sonst kommt eine zweite Rechnung ....
Die Seite ooo-81 enthält die aktiven Mitglieder - Stand Ostern 81 - :
Wer ein HP-9845 Barcodeplottprogramm möchte kann es bei Detlev (2) erhalten, wer an einer Sammelbestellung teilnehmen möchte, der wende sich an Ulli (3) und wer ein Buch aus der Bibliothek ausleihen möchte, schreibe an Frank (117). Diese Zuständigkeiten sind unbedingt zu beachten, denn nur so ist ein Berücksichtigung der Anfrage etc. gewährleistet.
Auffallend ist, daß Mitglieder mit höherer Nummer ziemlich passive Fans sind: Rührt Euch ! Es gibt viel zu tun. Das häufigste Argument auf solche Aufrufe lautet "Ich bin ja noch Anfänger" o.ä., doch das lasse ich nicht gelten. Auf versch. Treffen von Clubmitgliedern untereinander stellte sich immer wieder heraus: Ein persönl. Gespräch hilft mehr, als ein dutzend Briefe. Daher folgt den Beispielen von Matthias Grabiak und Andreas Marktscheffel, die im Raum Frankfurt eine Ortsgruppe gründen wollen (121-81). Äußerst begrüßenswert ist auch die Absicht von Günter Lehrke (239), im Großraum Stuttgart eine Ortsgruppe zu gründen. ALSO: Wer im Umkreis von 1oo km um Stuttgart möchte mitmachen ? Meldet Euch beim Günter! (Ortsgruppengründungen und -mitteilungen stehen im prisma immer ausreichend Seiten zur Verfügung, falls Ihr das eine oder andere sinnvolle mitzuteilen habt !) Oliver

---

Ein Witz ?

Es ist schon fast ironisch: Da hocken drei handvoll Leutchen über ganz Deutschland verteilt und kommen plötzlich auf die Idee, aus ihrem Club einen eingetragenen Verein zu machen. Selbst eine Einladung mit extrem dürftiger Resonanz wird verschickt .... und dann ? Die Vorbereitungen laufen auf Hochtouren, da man vereinzelt meint, es geht nicht ohne e-Punkt V-Punkt hinter dem Namen, man sucht Gründungsmitglieder aus ganz Deutschland, da ohne e.V. das Chaos drohen soll und dann auf einmal kommt das große BUMM-PENG und alles ist aus: Bei der Konsultation eines Rechtsanwaltes im Rahmen der Gründungsvorbereitungen trifft dieser die Feststellung, daß wir ohne e.V. viel besser fahren und rät energisch von dem Vorhaben ab ?!! Wir haben kein großes Vereinsvermögen, das Risiken mit sich bringt und erst recht keine Verpflichtungen irgendwem gegenüber; die Beiträge sind steuerfrei und keineswegs gigantisch hoch; der Club erwirtschaftet keine Gewinne, denn er ist gemeinnützig tätig: Was also soll der e.V. ??? Ein Fachmann lieferte auch diese Antwort: Mit dem Verein kommt eine Fessel für die aktiven Mitglieder. Eine auf Jahre hinaus gültige Satzung muß her, Vorstand, Vollversammlungen, Vorstandssitzungen und den aktiven Mitgliedern werden die Hände betreffs ihrer bisherigen Fähigkeit, sich auf Neuerungen einzustellen,gebunden. Die Folgen wären noch schlimmer als im MICAC, der bekanntlich mehr oder weniger eingeschläfert wurde.
Sehen wir uns einmal um in der Clubwelt der Computerclubs: Gibt es unter den Clubs, die nicht wirtschaftlich tätig sind (also nicht Apple-Club, SCC u.a.) auch nur einen e.V. ? Nein ! Nicht einmal der PPC ....
Also kam letzte Woche beim HP-Meeting der endgültige Beschluß: Wem unser Club nicht gefällt, der soll einen e.V. gründen, wo er will, aber ohne uns!

Eure "Aktiven"

Im November 1980 wurde der MEMORY-SAVER von Sven Beiersdorf veröffentlicht. Da er jedoch auf Matrix geschrieben war, haben ihn nur die ersten 150 Mitglieder erhalten. Für die anderen 350 sei er nachfolgend noch einmal wiedergegeben:

M E M O R Y S A V E R von Sven Beiersdorf

Zur Erzeugung des Effektes gehe man wie folgt vor:

1) beliebige Version eines Key-Assignment Programms einlesen
2) XEQ"KA"
3) Falls erforderlich auf die PAIRS Abfrage "1" eingeben: 1 R/S
4) Auf die erste Abfrage eingeben: 241 ENTER 65 ENTER 11 R/S
5) Auf die zweite Abfrage eingeben: 241 ENTER 56 ENTER 12 R/S
6) Evtl. ertönt BEEP (je nach KA Version). KA-Prgr. muß enden
7) Programm löschen: CLP"KA"
8) PRGM (In den Program-Modus schalten)
9) RCL 04
10) SST
11) PRGM (Aus dem Program-Modus aussteigen)
12) Taste 11 (Summe plus) im User-Modus drücken
13) PRGM (In den Program-Modus schalten)
14) SST ausführen (warten!)
15) SST ausführen (warten!)
16) Clear (Löschtaste) drücken und warten !
17) Clear drücken und warten
18) PRGM (Aus dem Program-Modus aussteigen)
19) Rechner ausschalten !!!!
20) Versuchen, den Rechner wieder einzuschalten ! Nicht möglich !

Nach dem Ausschalten des Rechners ist es nicht möglich, diesen wieder (auf die "normale" Art) einzuschalten oder MEMORY-LOST (Master-Clear) auszuführen.

Achtung! Im Interesse des Clubs bitte nicht weitersagen:

a) Der Rechner läßt sich durch kurzes Herausnehmen der Batterien und anschließendem Drücken der Taste CLEAR wieder einschalten. Der Effekt bleibt jedoch erhalten! Probiere es aus ...

b) Da es u.U. zu erheblichen Verzögerungen beim Bewegen per SST oder BST im Prgm-Modus kommen kann (und andere Gags) empfiehlt es sich nicht, diesen Zustand als permanent einzurichten. Er ist jedoch optimal geeignet, um unerwünschten Spielern schnell die Lust am Rechner zu vertreiben.

Die einzige Möglichkeit, MEMORY LOST zu verursachen, ist entweder das Einlesen einer halben WALL-Karte und anschließendem Drücken von CLEAR oder das Abspeichern eines undefinierten Wertes nach c, Auch stundenlanges Herausnehmen der Batterien dürfte zum Erfolg führen.

Alles auf eigene Gefahr ! (Wie üblich !!?!!) Sven Beiersdorf und Oliver Rietschel

---

Einfach ausschalten

Heinz Elsässer hatte das Problem, beim Ausschalten des Rechners das gleichzeitige Ausschalten des Druckers zu vergessen. Er schrieb daher folgendes Programm: LBL"OF" FS?21 BEEP OFF END. Das Programm hatte aber den Nachteil, nur zu testen, ob der Drucker angeschlossen ist, nicht aber, ob er auch eingeschaltet ist. Daher schrieb ich nachfolgende Routine, die allerdings den Nachteilhat, das Alpharegister bei angeschlossenem Drucker zu zerstören. Freilich kann man

das Programm dahingehend abwandeln, daß das Alpharegister erhalten bleibt, doch wird dann die Meldung "PRINTER ON !" nicht gegeben und der Alpharegisterinhalt bei angeschlossenem Drucker ausgegeben. Hat jemand noch einen Verbesserungsvorschlag ??

```
LBL"OF" FC? 55 OFF CLA SF 25 PRA FC?C 25 OFF CF 21 "PRINTER ON !"
AVIEW SF 21 CLA BEEP BEEP PSE OFF END
```

Oliver (1) und
Heinz (47)

Insbesondere neuere Mitglieder fragen vereinzelt an, ob für Veröffentlichungen oder eingesandte Programme eine Entschädigung gezahlt wird:
Durch eine 198o getroffene Konvention wurde die Sache folgendermaßen geregelt: Für eingesandtes Material wird kein Honorar gezahlt. Vielmehr wird im Oktober ein Wahl durchgeführt, in der die besten Veröffentlichungen mit versch. Preisen honoriert werden sollen. DIese Wahl wird stattfinden.

Mittelschwere Resonanz
erfuhr meine Anmerkung in prisma 26-81 zu dem Programm von Gerhard. Dort schrieb ich ziemlich lapidar "R/S läßt sich doch jeder beliebigen Taste zuordnen ...". Das stieß bei einigen Clubmitgliedern auf erhebliche Schwierigkeiten, so daß hier die Antwort folgen soll:

Zuordnen von R/S zu einer beliebigen Taste des HP-41c(v) Tastenfeldes
R/S läßt sich nicht über die ASN-Funktion des 41 zuordnen. Vielmehr muß man auf eine beliebige Version des KA-Programms zurückgreifen, wobei die Anwendung aber denkbar einfach ist: Statt Byte1 und 2 des betreffenden 2-Byte-Befehls gibt man einfach 5ENTER,ENTER,xy ein, wobei xy die betreffende Taste bezeichnet. Beispiel: XEQ"KA" - es erscheint die Abfrage PRE/POST/KEY (je nach Version) - Eingabe: 5 ENTER ENTER 11 - R/S drücken - Sofern das Prgr. eine zweite Eingabe verlangt, diese tätigen - fertig; R/S ist jetzt der Taste 11 (Summe +) zugeordnet.

Wen dieses Gebiet der synthetischen Programmierung interessiert, der sollte aufmerksam die Rechnerorganisationsberichte in prisma lesen und sich das Buch Synthetic Prgr. aus der Clubbibliothek ausleihen. Oder an einem Clubtreffen teilnehmen.... Oliver (1)

PPC - ROM Ade ? PPC - ROM Good Bye ?

Nur ältere Mitglieder (niedr.Mitgl.Nr. unter 15o) werden überhaupt näheres über das PPC-ROM wissen, das sich unser Club in sechsfacher Ausführung bestellt hat und verleihen will. Es wird vom amerikan. PPC produziert und soll eine Vielzahl nützlicher Programme, zum großen Teil synthetische, enthalten. Allerdings gibt es nicht nur bei unserem Amateurclub organisatorische Schwierigkeiten - auch der PPC hat sie. Das Produktionsdatum - ursprünglich auf Mitte März 81 festgelegt - hat sich bereits bis September verschoben. Und wer weiß, ob es dann auch wirklich kommt .... ?
Wie dem auch sei - wenn die Dinger endlich kommen, wird es im prisma zu lesen sein. Wenn .... Oliver (1)

Gerade jetzt, wo dieses prisma fast fertig ist, bekomme ich einen Anruf vom Walter (2o5), der noch eine Anzeige für die Clubbörse hat. Doch die ist für heute leider voll, weswegen - man möge es mir verzeihen - ich sie in diese letzte freie Ecke klatsche:

Oliver Rietschel
Hewlett-Packard Anwender-Club
Postscheckkonto Hmb. 48835-207
Postfach 373 · D-2420 Eutin

Clubmitglieder können kostenlos Gelegenheitsanzeigen in der Clubbörse aufgeben !

Zusätzlich zu den in prisma 65-81 aufgeführten Leistungen biete ich jetzt an:
1) Umbau von zei ROMs in ein Double ROM bzw. Einbau von ROMs in den Rechner
   Umbau: 4o,-- Eibau: 45,-- pro ROM
2) Modifizierung von Quad-RAMs:
   a) Eibau von Kondensator: 4o,-- pro RAM
   b) Einbau von 3 (evtl. 4) Quad-RAMs in das Gehäuse des 41c. Mit Schalter werden die Quads ein- bzw. ausgeschaltet: Beim 41 cv nicht möglich !!
   Eibau des 1.Quad: 12o,-- jedes weitere: 1oo,--

Die Module und evtl. Rechner sollten mir per Wertbrief zugesandt werden (nach Vereinbarung). Die Leistungen beinhalten den gewünschten (auch Sonderwünsche wie z.B. bestimmte Fortbelegungen einzelner Module) Umbau mit Gewährleistung der Funktionstüchtigkeit. Zahlungsweise: Vorraus Bar oder Scheck
Die Arbeiten werden innerhalb von 2 Tagen durchgeführt, so daß die zugesandten Teile spätestens am 3. Tag nach Erhalt zurückgeschickt werden.
Hans-Günter Lütke Uphues, Bahnhofstr. 2o, 44o1 Altenberge, Deutschland

Nachfolgend gebe ich eine verkürzte Gerätebeschreibung für ein Kassetten-Interface für den HP-41 c(v) wieder, sowie eine kurze kritische Stellungnahme:

KASSETTENINTERFACE FÜR HP-41C:
Zweck: Ermöglichung der Speicherung von Programmen, Daten und Statusinformationen auf Magnetfolien-Massebspeichern, in der Regel also Kassetten- oder Tonbandgeräte, zum späteren Einlesen in den HP-41c(v). Das Einspeichern geschieht mittels dem opt. Lesestift, indem entsprechender Strichcode abgetastet wird. Hervorzuheben ist:
- Es lassen sich alle nichtsynthetischen Programme eingeben, auch wenn hierzu kein strichkodiertes Programmlisting existiert, indem die Einzelfunktionsstrichcodes des Barcodetastenfeldes benutzt werden.
- Es können nur konstante Datensätze in den Rechner eingelesen werden. Ein Auslesen von Daten aus dem Rechner ist nicht möglich.
- Zum Betrieb des Interfaces wird die Betriebsfirmware des Lesestiftes in vollem Umfange verwendet; man kann deshalb beim Arbeiten mit dem Interface sämtliche Funktionen des Lesestiftes aufrufen (WNDDTA,WNDDTX,WNDLNK,WNDSUB,WNDSCN,WNDTST)
- Die Möglichkeit der gemischten Verwendung von Programm-, Daten-, und Einzelfunktionsbarcodes ermöglicht es, einem auf Band abgelegten Programm unmittelbar dazugehörige Statusinformationen und Konstanten beizufügen, was z.B. beim Magnetkartenleser nicht möglich ist.

Das Interface kann durch Erweitern mit einem Modem zur Datenfernübertragung per Telefon oder Funk umgerüstet werden.

Es sind keine Erweiterungen oder sonstigen Veränderungen am Rechner nötig. Das Interface wird lediglich in die Zuleitung zum optischen Lesestift eingeklemmt.

Lieferumfang: Das Gerät wird auf Europakarte (1oo x 16o mm) mit indirekter 31-poliger Steckerleiste nach DIN 41617 mit detaillierter Beschreibung geliefert. Der Preis beträgt 266,--DM, für Clubmitglieder unter Angabe der Mitgliedsnummer allerdings nur 19o,--DM !!! Mit dem Kauf dieses Gerätes spart ein Mitglied also Geld in fast doppelter Höhe des Clubbeitrags !!!
Das Gerät sowie eine etwas ausführlichere Beschreibung (kostenlos geg. Rückporto vorab) ist erhältlich bei Clubmitglied 283: Ulrich Jansen, Mühlenstr. 16
514o Erkelenz, Telefon: o2154-5874

Kritische Stellungnahme: Das Gerät versucht zweifelsohne in eine Marktlücke vorzustoßen. Der Bedarf und die Nachfrage nach einem solchen Gerät sprechen dafür, denn Magnetkarten sind keineswegs optimal. Eines darf man aber nicht übersehen: Programme können nur soschnell ausgeführt und/oder eingelesen werden, wie man sie selbst beim ersten Mal eingibt. Das ist keineswegs schnell ... man braucht viel Zeit!

Nachdem mit dem ersten Lösungsbuch einige Erfahrungen gesammelt worden sind, erscheint nun das zweite Lösungsbuch, an dem zwei Autoren beteiligt sind: Werner Loibl (224) und Walter Kropf (56). Das Buch stellt qualitativ die Spitze dessen dar, was ein Lösungsbuch auf diesem Sektor überhaupt bieten kann. Besonders ist der Teil von Werner hervorzuheben, der nicht nur sehr schöne Programme geschrieben hat, sondern in einer lobenswerten und mühsamen Arbeit alle theoretiscehn Hintergründe der Programme ausführlich behandelt.
Die Programme wurden von Niels Nöhren (81) auf ihre Richtigkeit hin überprüft und in Zusammenarbeit mit den Autoren weiter verbessert. Die relativ hohen Autorenhonorare, die im Preis enthalten sind, sind dadurch mehr als gerechtfertigt. Das 63 DINA4 Seiten starke Buch wird in limitierter Auflage (ca. 5o) erst nach Eingang der Bestellungen gedruckt. Daher sofort bei Oliver bestellen!!! Evtl. wird das Buch noch durch versch. Biorhytmusprogramme ergänzt.

Daten: Mindestens 63 "echte" DINA4 Seiten, unverkleinert, da sehr kleine Schriften verwendet.
Preis: je nach Nachfrage und Auflage zwischen 4o,-- und 5o,--DM
Zahlung: Nach Erhalt der Rechnung
Haftung und Garantie: Es gelten die normalen Clubregeln.

Inhalt:
1) Datum beinhaltet versch. Kalenderrechnungen im gregorianischen und julianischen Kalender, manche Berechnungen für jedes beliebige vor und nachchristliche Datum innerhalb der Anzeigekapazität des HP-41c(v).
2) Feiertage: berechnet den Wochentag datungebundener und das Datum Osterabhängiger Tage für nachchristliche Jahre im gregorianischen und julianischen Kalender.
3) Zeiträume lassen sich ohne Zwischenschaltung einer Kalenderroutine durch direkte Subtraktion bzw. Addition der Differenzen zwischen Tagen finden oder Zeiträume berechnen, zum Datum addieren oder davon abziehen. Funktioniert mit beliebigen vor- und nachchristlichen Daten innerhalb der Anzeigekapazität des Rechners.
4) Wiederkehr berechnet gregorianisch und julianisch, jedoch nur nachchristlich, beliebig viele aufeinanderfolgende Jahre, an denen ein Wochentag auf ein bestimmtes Datum fällt, wann also z.B. der 24.Dez. ein Mittwoch ist (war).
5) Kalender druckt einen Kalender aus, allerdings im Gegensatz zu dem in prisma bereits veröffentlichten Programm auch nach dem julianischen Kalender und Berücksichtigung der Tatsache, daß seit 1975 die Woche mit dem Montag beginnt.
6) Verschiedene nützliche Unterprogramme.
7) Wissenswertes zum Kalender:
   - Julianischer Kalender
   - Gregorianischer Kalender
   - Christliche Zeitrechnung
   - Stile
   - Jüdischer Kalender
   - Mohammedanischer Kalender
   - Hundertjähriger Kalender
   - Das Jahr
   - Der Monat
   - Jahreszeiten
   - Kalenderreform
   - Zeitgleichung
8) Kalenderrechnungen versch. Art, auch als Unterprogramm.
9) Bewegliche Feste: Über die goldene Zahl des eingegebenen Datums werden die bewegl. Feste berechnet, die in diesem Jahr liegen.
1o) Monatskalenderberechnung in Matrizenform
11) Jahreskalenderberechnung in Listenform
12) Biorhytmus Vers.1.1.
13) Freitag der 13. berechnet jeden Freitag, der auf den 13. eines Monats fällt.
14) Biorhytmus Vers. 2.1.

Autoren: 1-7: Werner Loibl, 8-12: Walter Kropf, 13-14: Niels Nöhren

PROGRAMMIERHILFEN FÜR ANFÄNGER

Von Hans-Günther Götter (115) stammt ein Programm zur Textverarbeitung, an dem sich einige wesentliche Prinzipien dieser für den HP-41 C(V) charakteristischen Betriebsart demonstrieren lassen. Das Programm dient zum Schreiben und Lesen eines (fast) beliebig langen Textes auf dem Display des Rechners. Das Schreiben wird durch XEQ WR und das anschließende Lesen durch XEQ RD ausgelöst. Nach Eingabe von bis maximal 24 Alphazeichen wird R/S gedrückt, und weitere 24 Zeichen können eingegeben werden. Wenn die Eingabe beendet ist, wird R/S abermals gedrückt; Ende wird durch BEEP angezeigt.

```
01♦LBL "WR"
02 CLRG
03 .003
04 STO L
05 SF 00
06 AON
07♦LBL 00
08 "TEXT?"
09 FC?C 00
10 "├ CONTI
N."
11 CF 23
12 STOP
13 FC? 23
14 GTO 04
15♦LBL 01
16 ASTO IND
L
17 ASHF
18 ISG L
19 GTO 01
20 .004
21 ST+ L
22 GTO 00
23♦LBL "RD"
24 .003
25 STO L
26 AON
27 CLA
28♦LBL 02
29 CLST
30 ARCL IND
L
31 X<> IND
L
32 X=Y?
33 GTO 04
34 X<> IND
L
35 ISG L
36 GTO 02
37 .004
38 ST+ L
39 AVIEW
40 CLA
41 GTO 02
42♦LBL 04
43 BEEP
44 AOFF
45 END
```

Zunächst zur Texteingabe (Programmteil "WR"): Auf die Frage "TEXT ?" (im weiteren Programmlauf heißt es dann "TEXT ? CONTIN.", was durch Setzen -05-, Abfragen und Löschen -09- von FLAG 00 bytesparend erreicht wird) muß der gewünschte Text (max. 24 Zeichen/Zeile) eingegeben werden. Dazu muß der Rechner durch AON -06- zunächst einmal für ALPHA-Zeichen empfangsbereit gemacht werden, andernfalls würde er jeden Tastendruck als Ziffer oder Funktion (in diesem Falle sinnlos) interpretieren. Der gleiche Befehl ist allerdings im zweiten Programmteil -26- überflüssig, da Ausgaben wegen AVIEW -39- immer im richtigen Modus erfolgen; es ist allerdings im Interesse nachfolgender Programme oder manueller Rechnungen vorteilhaft, den ALPHA-Modus am Ende durch AOFF -44- wieder aufzuheben.

Der eingegebene Text wird nach jeder Zeile (Drücken von R/S) in Blöcken von jeweils 6 Zeichen fortlaufend ab R00 abgespeichert. Das besorgt die Schleife mit LBL 01 (Zeile 15 bis 19), wobei der Index des Zielspeichers platzsparend im LAST X-Register steht und dort auch jeweils von 0 bis 3 bezw. von 4 bis 7 etc. inkrementiert wird -18-. Natürlich muß der im ALPHA-Register stehende Text durch ASHF -17-

jeweils nach dem Abspeichern eines 6er-Blocks um 6 Zeichen nach links geschiftet werden. In Zeile 20 wird der Endwert der Laufvariablen im LAST X-Register nach Abspeichern der ganzen Zeile jeweils um 4 erhöht, worauf zu Zeile 7 zurückgekehrt wird, um nach der nächsten Textzeile zu fragen.

Die Eingabe wird beendet durch nochmaliges Drücken von R/S nach einer Texteingabe. Was geschieht dabei? Bevor das Programm in Zeile 12 zur Aufnahme eines neuen Textes stoppt, wird in Zeile 11 FLAG 23 gelöscht. Durch Eingabe des neuen Textes wird es automatisch wieder gesetzt (ALPHA-Zeichen-Eingabe-Flag, Handbuch, Seite 217), und bei der Abfrage in Zeile 13 wird daher der Sprungbefehl "GTO 04" ignoriert. Nur in dem Falle, wo keine neue Texteingabe nach dem STOP -12- erfolgt (Drücken von R/S nach der Aufforderung zur Texteingabe), bleibt FLAG 23 gelöscht, und das Programm verzweigt nach Zeile 42, wo es mit einem BEEP endet. Die gleiche Prozedur wird übrigens häufig zur Beendigung einer Serie von Zifferneingaben benutzt, dann allerdings mit FLAG 22 anstelle von 23.

Die Textausgabe (Programmteil "RO") ist sehr ähnlich aufgebaut, enthält aber noch zwei in diesem Zusammenhang interessante Punkte: zunächst einmal werden immer maximal 4 6er-Blocks von Zeichen aus den Speicherregistern ins ALPHA-Register geholt. Da das mit dem Befehl ARCL... -30- geschehen muß, der die Eigenschaft hat, den gerufenen Speicherinhalt immer an das bereits im ALPHA-Register stehende anzuhängen, muß dieses zu Anfang gesäubert werden: 27 CLA. Auf diese ALPHA-Löschung sollte man immer achten, denn es kommt häufig vor, besonders bei Ergebinsanzeigen (z.B. "25.4 MM"), daß das ALPHA-Register mit ARCL... geladen wird.

Der Ausstieg aus der Textanzeigeschleife geschieht durch die Zeilen 31 bis 34: Da zu Anfang der Texteingabe alle Register gelöscht worden sind -02-, und da auch wegen 29 der gesamte Stack auf 0 gesetzt worden ist, wird in diesen Zeilen geprüft, ob der gerufene Registerinhalt 0 ist. In diesem Fall ist man sicher, das erste Register erwischt zu haben, das keine ALPHA-Zeichen des eingegebenen Textes enthält, und das Programm darf nach Zeile 42 verzweigen, wo es wie oben mit einem BEEP endet. Allerdings hätte sich dieser Ausstieg auch etwas kürzer durch LBL 02; ARCL IND L; 0; RCL IND L; X=Y?; GTO 04; ISG L lösen lassen.

Klaus Werner Hoenow

```
 01•LBL "UEBE"
 02•LBL 01
CF 29  FIX 0
"MAX ZAHL ?"  PROMPT  1
+  STO 05  STO 00  5
"ZEIT ? SEC"  PROMPT
STO 06  SIN  ST* 00
"FALSCH"  ASTO 12
"LAMA"  ASTO 13  "KAMEL"
ASTO 14  XEQ 08  STO 02
XEQ 08  STO 03

 27•LBL 02
0  STO 09  STO 10  10
STO 08  "+,-,*,/?"  AON
PROMPT  AOFF  ASTO 11

 38•LBL 03
RCL 03  RCL 02
XEQ IND 11  STO 04  CLA
ARCL 03  ARCL 11
ARCL 02  "├="  AVIEW
XEQ 08  STO 02  XEQ 08
STO 03  RCL 06  STO 07
CF 22  CF 00  "├?"
AVIEW

 59•LBL 04
PSE  FS? 22  GTO 06
DSE 07  GTO 04  VIEW 13

 66•LBL 05
TONE 2  PSE  ,1  ST+ 10
FS?C 00  GTO 00  SF 00
RCL 06  STO 07  CF 22
AVIEW  GTO 04

 79•LBL 00
VIEW 14  PSE  "├="
ARCL 04  AVIEW  PSE
GTO 07

 87•LBL 06
RCL 04  X=Y?  GTO 00
VIEW 12  GTO 05

 93•LBL 00
"RICHTIG"  AVIEW  1
ST+ 09

 93•LBL 07
DSE 08  GTO 03  RCL 10
X=0?  XEQ 10  RCL 09  ,1
/  CLA  ARCL X
"├% RECHT"  AVIEW  PSE
GTO 02
```

Dieses UEBE-Programm ist eine Weiterentwicklung des "ARITHMETIK-LEHRGANG"-Programms aus der Standard-Programmsammlung. Vor allem wird für jede Aufgabe eine (zuvor wählbare) Rechenzeit vorgegeben, und die eingetippten Rechenergebnisse werden automatisch abgeholt (ohne RS).

Funktion des (XEQ) "UEBE"-Programms:

⇒ Der Rechner fragt nach der "MAX ZAHL ?", die der Zufallgenerator für die Summanden verwenden darf: z.B. 30, RS.

⇒ Der Rechner fragt nach der "ZEIT ? SEC ?" für jeweils eine Aufgabe. Zahl, RS. (Wenn nur RS ohne Zahl gedrückt wird, setzt der Rechner von sich aus 5 Sekunden je Aufgabe.)

⇒ Der Rechner fragt nach der Rechenart "+,-,•,:?". ■+ oder ■- oder ■• oder ■: eingeben, RS.

⇒ Der Rechner stellt nacheinander 10 Aufgaben in der üblichen Schreibweise, z.B. "12+23=?". Ergebnis eintippen, z.B. 35 (kein RS!, die Zahl wird sogleich automatisch abgeholt).

⇒ Der Rechner meldet "RICHTIG" und stellt die nächste Aufgabe, z.B. "14+8=?". Ergebnis eintippen, z.B. 32.

⇒ Der Rechner meldet "FALSCH", piept unangenehm und stellt die gleiche Aufgabe zum 2. Mal. Ergebnis eintippen, z.B. 9.

⇒ Der Rechner zweifelt an Ihrer Ernsthaftigkeit, meldet sich mit "KAMEL" und piept wieder unangenehm. Um keine Zeit zu verschwenden, zeigt der Rechner die richtige Lösung: "14+8=22" und stellt schließlich die nächste Aufgabe, z.B. "28+29=?".

Was, Sie brauchen länger, als die von Ihnen vorgegebene Zeit???

⇒ Der Rechner piept ärgerlich und sagt so etwas wie "LAMA" und stellt die Aufgabe zum 2. Mal.

⇒ Nach 10 Aufgaben sagt der Rechner, wieviel % der gestellten Aufgaben richtig waren.

⇒ Sollten alle 10 Aufgaben auf Anhieb richtig gelöst worden sein, belohnt Sie der Rechner mit einer kleinen Melodie (-wenn man sie wegläßt, ist das Programm kürzer-) und fragt, ob man nochmal "+,-,•,:?" mit gleichen Bedingungen rechnen will.

⇒ Von ganz vorne? (EXQ) Σ+ (=XEQ 01).

Bemerkungen [illegible] Vergleichsoperationen

Bei Vergleichsoperationen, beispielsweise: x = y? entstehen mitunter Überraschungen, die mit der Rechengenauigkeit (Rundungstechnik) der HP-Taschenrechner zu erklären sind.

Dazu wird Seite 78 des Handbuchs in Erinnerung gerufen. Im Kapitel "Rundung einer Zahl" heißt es u.a.:

> " --- wird bei Änderung des Anzeigeformats --- die Genauigkeit der internen Darstellung der Zahlen im Rechner nicht beeinflußt. Dort sind alle Zahlenwerte mit zehn --- Stellen in der Mantisse --- gespeichert."

Einige Beispiele, auszuführen mit FIX 9

Berechne 441 : 19 und multipliziere das Resultat mit 19

441, ENTER, 19, ./.; Resultat: 23,210 526 32 * 19, Resultat: 441,000 000 1

Berechne LN 441 und führe anschließend $e^{(LN\ 441)}$ aus

441, LN; Resultat: 6,089 044 875, e^x; Resultat: 440,999 999 8

Berechne 13^7

13, ENTER, 7, y^x; Resultat: 62 748 517,00

Berechne $\sqrt[7]{62\ 748\ 517}$

a) 62 748 517, LOG, 7, ./., 10^x; Resultat: 12,999 999 99

b) 62 748 517, ENTER, 7, 1/x, y^x; Resultat: 13,000 000 01

Die Algorithmen sind korrekt, die Ergebnisse in der Anzeige keinesfalls. Korrekte Anzeigen erhält man allerdings sofort bei Umschaltung auf FIX ≤ 7, was allerdings nichts daran ändert, daß intern weiterhin mit den Ergebnissen gemäß a) und b) operiert wird.

Wollte man also beispielsweise die Vergleichsoperation: x = y? mit x = 13 und $y = \sqrt[7]{62\ 748\ 517}$ durchführen, käme dieser Vergleich nicht zu Stande, denn intern liegen die erwähnten, von "13" abweichenden Ergebnisse vor. Abhilfe ist nur durch Anwendung der oben erwähnten "Rundung" [RND] möglich. Einzelheiten siehe Handbuch.

G.SIEWERT (52)

```
113•LBL 08
RCL 00  PI  *  FRC
STO 00  SQRT  RCL 05  *
INT  RTN

124•LBL "+"
+  RTN

127•LBL "-"
-  X≠0?  RTN  CHS
RCL 02  X<> 03  STO 02
RDN  RTN

137•LBL "*"
*  RTN

140•LBL "/"
X=0?  ISG 02  FIX  CLX
RCL 02  ST* 03  RDN  RTN

149•LBL 10
TONE 8  TONE 9  SIN
TONE 8  TONE 8  TONE 8
TONE 7  TONE 8  TONE 8
TONE 7  TONE 8  TONE 9
SIN  TONE 9  TONE 8  SIN
TONE 8  TONE 7  SIN
TONE 7  TONE 3  RTN
.END.

LBL 'UEBF
.END.          330 BYTES
SIZE 16   HAUKE TETENS
```

Ec[illegible]rd Müller (122)
M.-Schnabrichstr.9
3500 KASSEL
tel:0561/492266

```
 01◆LBL "NEW
TON"
 02 "FUNKT.
NAME ?"
 03 AON
 04 PROMPT
 05 AOFF
 06 ASTO 03
 07 FIX 9
 08◆LBL A
 09 "X0 ?"
 10 PROMPT
 11 STO 01
 12◆LBL 00
 13 RCL 01
 14 STO 00
 15 XEQ IND
03
 16 STO 02
 17 1 E-4
 18 ST+ 01
 19 RCL 01
 20 XEQ IND
03
 21 RCL 02
 22 -
 23 1 E-4
 24 /
 25 RCL 02
 26 X<>Y
 27 /
 28 CHS
 29 RCL 00
 30 +
 31 STO 01
 32 VIEW X
 33 RCL 00
 34 X≠Y?
 35 GTO 00
 36 BEEP
 37 END

            CAT 1
LBL'NEWTON
END        79 BYTES
.END.      05 BYTES
```

NEWTON - VERFAHREN zur Bestimmung
von reellen Nullstellen einer Funktion

Voraussetzung

Unter einem globalen Label muß eine
Gleichung f abgespeichert sein.
Beim Aufruf von f ist der aktuelle
x-Wert im x-Reg. ausserdem in Reg.01.
Es muß von f eine reelle Lösung existieren
(ist das nicht der Fall, oder gibt es
zu viele Lösungen, z.B. y=sin x, so
sieht man nach einiger Übung die Diver-
genz der x Werte, da der aktuelle x Wert
jeweils angezeigt wird), außerdem muß
f zweimal differenzierbar sein und f'
darf in einer Umgebung der Nullstelle
nicht Null sein.

Wie das Newtonsche Verfahren funktioniert,
möge der, der's wissen will bitte nachlesen,
es steht in (fast) jedem Schulbuch!

Beispiel

Wurzel aus 2 soll bestimmt werden.

```
LBL'TEST         XEQ'NEWTON          FUNKT.NAME?
x↑2
2                TEST       R/S       X0 ?
-                2          R/S      aktuelles x
end                                  BEEP
                                     Nullstelle
```

Neustart mit einem anderen x0 möglich
durch XEQ A ($\Sigma+$)
x0 sollte möglichst nah an einer vermuteten
Nullstelle liegen

size=004

Lineare Regression mit Fehlerrechnung

```
 01♦LBL "LINF"
CLRG  SF 21  ΣREG 04
"GIVE DATES"  AVIEW
RCL 08  RCL 04  RCL 06
RCL 09  /  *  -  RCL 07
RCL 06  X↑2  RCL 09  /
-  STO 00  /  STO 03
MEAN  X<>Y  RCL 03  *  -
STO 02  "a="  ARCL X
AVIEW  "b="  ARCL 03
AVIEW  RCL 07  RCL 03
X↑2  *  RCL 05  RCL 09
RCL 02  X↑2  *  +  +
RCL 04  RCL 06  RCL 03
*  -  RCL 02  *  RCL 03
RCL 08  *  +  ST+ X
RCL T  X<>Y  -  RCL 09
2  -  /  SQRT  STO 01
"S.Y="  ARCL X  AVIEW
RCL 01  RCL 07  RCL 00
RCL 09  *  /  SQRT  *
"S.a="  ARCL X  AVIEW
RCL 01  RCL 00  1/X
SQRT  *  "S.b="  ARCL X
AVIEW

 09♦LBL 01
"X=?"  PROMPT  RCL 03  *
RCL 02  +  "Y="  ARCL X
AVIEW  GTO 01  .END.
```

Formeln:

$$b = \frac{\sum x_i y_i - 1/n \sum y_i \sum x_i}{\sum x_i^2 - 1/n \ (\sum x_i)^2} \qquad \text{Steigung}$$

$$a = \bar{y} - b \cdot \bar{x} \qquad \text{Achsenabschnitt}$$

Fehler (Standardabweichung)

Fehler des Fkt.wertes:

$$s_y = \sqrt{\frac{\sum (y_i - (bx_i + a))^2}{n - 2}}$$

Fehler der Steigung:

$$s_b = \sqrt{\frac{n}{n \sum x_i^2 - (\sum x_i)^2}} \cdot s_y$$

Fehler des Achsenabschnittes:

$$s_a = \sqrt{\frac{\sum x_i^2}{n \sum x_i^2 - (\sum x_i)^2}} \cdot s_y$$

Programmumfang: 151 Bytes

SIZE 010

Zum Gebrauch des Programms diene flg. Beispiel:

Geg. sind die Werte

x	1	2	3	4
y	5	8	9	10

1) XEQ' LINF
2) Nach der Aufforderung "GIVE DATES", Eingabe der Werte wie folgt:
   1;ENTER;5;Σ+ , 2;ENTER;8;Σ+ , etc.
3) R/S: a=4,00 ; R/S: b=1,60 ; R/S: S.Y=0,77 ; R/S: S.a=[illegible],95
   R/S: S.b=0,35
4) R/S: X=? Um die Reg.gerade zu zeichnen, kann man sich einige Fkt.werte ausrechnen lassen, z.B.: 1 R/S Y=5,6; R/S: X=?
   4 R/S Y=10,4

Eine Korrektur erfolgt wie unter 2), aber statt Σ+, Σ- drücken.

Gerald Krampe

Schnelle PRIMFAKTORZERLEGUNG mit dem HP 41C

```
PRP "PFZ"

01•LBL "PFZ"
"PRIM? ZAHL=?" PROMPT

04•LBL 01
FIX 0  CF 00  SF 11
CF 21  STO 01  STO 04
INT  LASTX  "ZAHL WAR 0"
X=0?  GTO 05
"ZAHL WAR NEG."  X<0?
GTO 05  " Z. WAR BRUCH"
X≠Y?  GTO 05  1  -
"ZAHL WAR 1"  X=0?
GTO 05  RCL 04  SQRT
STO 03  "F.="  2  STO 05
XEQ 03  3  STO 05
XEQ 03  1  STO 05

39•LBL 02
4  ST+ 05  XEQ 03  2
ST+ 05  XEQ 03  GTO 02

47•LBL 03
RCL 03  RCL 05  X>Y?
GTO 04  RCL 04  RCL 05
MOD  X≠0?  RTN  SF 00
ARCL 05  "⊢,"  AVIEW
RCL 04  RCL 05  /
STO 04  SQRT  STO 03
GTO 03

68•LBL 04
ARCL 04  "⊢."  FS? 00
GTO 05  CLA  ARCL 01
"⊢=PRIM"

76•LBL 05
AVIEW  TONE 9  FS? 55
XEQ 06  STOP  GTO 01

83•LBL 06
SF 21  RCL 01  ACX  125
ACCHR  32  ACCHR  ACA
PRBUF  END

LBL'PFZ
END          213 BYTES
```

RECHENZEIT:
$T \approx 0.003 \cdot \sqrt{Y} + 0.04$ [MIN]

Y = größte in der Zahl vorkommende Primzahl

Sämtliche mir bekannten Programme für Primzahlen bzw. zur Primfaktorzerlegung sind eine grauenhaft furchtbare Zumutung, das stundenlange Warten insbesondere bei großen Zahlen ist eine völlig sinnlose Geduldsprobe und tötet einem die letzten Nerven!!! Daher habe ich so zwischendurch ein unwahrscheinlich viel schnelleres Programm, das zudem noch einigen Komfort bietet, erarbeitet. Dieses neue Programm zerlegt z.B. die Zahl 6 869 821 302 in nur 4 Minuten in seine Primfaktoren!!! Das ist nur 1,1% der Zeit, die das erst kürzlich im Sonderheft "Weihnachten 80" von Herrn Christof Born erschienene Programm für die gleiche Zahl brauchte, nämlich ca 6 Stunden! Vielleicht sind andere HP 41C-Besitzer auch an diesem schnellen Programm interessiert:

Nach dem Einlesen der Karte verlangt der Rechner im Klartext nach der zu untersuchenden Zahl, die sogleich auf Zerlegbarkeit geprüft wird, widrigenfalls erscheint Fehleranzeige im Klartext. Werden Faktoren gefunden, so werden sie alle in **e i n e r** Zeile übersichtlich dargestellt. Ist zusätzlich noch ein Drucker angeschlossen, so werden die AVIEW-Druckbefehle für den Drucker unwirksam gemacht und dafür am Schluß die eingegebene Zahl und daran anschließend die gesammelten Primfaktoren einzeilig ausgedruckt. Anleitung:

Nr.	Text	Tasten	Anzeige
1.	Programm einlesen (autostart)		PRIM? ZAHL=?
2.	Zahl eingeben:	"Zahl", RS	
	z.B.	90, RS	F.= 2,3,3,5.
	z.B	41, RS	41 = PRIM
	z.B.	15,6, RS	ZAHL WAR BRUCH
	z.B.	1, RS	ZAHL WAR 1
	z.B.	-7, RS	ZAHL WAR NEG.
	usw., usw., usw.		
*	mit Drucker: Druckerbild (MAN)	740, RS	740⊢ F.= 2,2,5,37.
*	bei Beginn mitten im PGRM statt RS: "Zahl", XEQ, 01		

Hauke Tetens, 7218 Trossingen, Fr.-Ebert-S.16

Vergleich zur Rechengeschwindigkeit Programm PFZ

Argument	Faktoren	Rechenzeit in sec (o. Drucker) Version Niels	Version Herbert/Jürgen
15987	3,73,73	22	16
1Ø24	$2^{1Ø}$	5	7
1ØØ9	Primzahl	11	7
2Ø18	2, 1ØØ9	255	9
1ØØØØ1	11, 9Ø91	24ØØ	22

Das letzte Beispiel ist wohl etwas extrem, aber vielleicht gerade deshalb interessant.

PFZ in unserer Version wird nach Eingabe der zu untersuchenden Zahl mit XEQ PFZ oder Zuordnung gestartet, für Zerlegung einer neuen Zahl einfach R/S hach Eingabe der Zahl.

Listing:

```
PRP "PFZ"

01•LBL "PFZ"
02 FIX 0
03 CF 00
04 STO 01
05 SQRT
06 STO 00
07 RCL 01
08 INT
09 LASTX
10 X≠Y?
11 GTO 03
12 2
13 STO 02
14 X>Y?
15 GTO 03
16 X=Y?
17 GTO 02
18 X<>Y
19 0
20 SF 02
21 X>Y?
22 SF 01
23 CLA

24•LBL 01
25 RCL 01
26 RCL 02
27 MOD
28 X=0?
29 GTO 04
30 FS? 01
31 GTO 02
32 FC?C 02
33 GTO 05
34 3
35 STO 02
36 GTO 01

37•LBL 05
38 2
39 ST+ 02
40 RCL 00
41 RCL 02
42 X<=Y?
43 GTO 01
44 FC? 00
45 GTO 02
46 RCL 01
47 ARCL X
48 GTO 00

49•LBL 02
50 "PRIMZAHL"
51 GTO 00

52•LBL 03
53 "UNZERLEGBAR"
54 GTO 00

55•LBL 04
56 SF 00
57 CF 01
58 RCL 02
59 ARCL X
60 ST/ 01
61 RCL 01
62 ENTER↑
63 SQRT
64 STO 00
65 RDN
66 1
67 X≠Y?
68 GTO 01

69•LBL 00
70 CF 00
71 CF 01
72 CF 02
73 AVIEW
74 TONE 9
75 END
```

Jürgen Moeck
Dellplatz 7 / 117
41oo Duisburg 1

Septemberausgabe 1981 - Fertigstellung: o1-o8-81 - Versand: 1o-o8- bis 17-o8-81

Liebe Mitglieder !

Die heutige Ausgabe ist wieder einmal was Quantität und Qualität anbelangt ein Superding geworden ! Besonders verweise ich auf den Barcodes mit HP-41 Standarddrucker und den Syntheticbericht für "Anfänger". Bei der Zusammenstellung hat mir diesmal Walter (2o5) geholfen. Mitleid sollten wir mit Detlev (2) haben, der diesmal nicht nur druckt und sortiert, sondern auch die Infos leimt und versendet ! Viel Spaß ! Oliver (1)

- Frankfurter Ortsgruppe: Unter Leitung von Andreas Marktscheffel wird am 1.August eine Frankfurter Ortsgruppe gegründet. Bericht im nächsten Info !

- Diesem Info liegen eine ganze Reihe von Frage und Bestellbögen bei. Bitte beachten !

- Das Lösungsbuch "Intelligente Spiele" ist fertig. Siehe Seite 286-81 !!

Kölner Ortsgruppe: Wer an einer Kölner Ortsgruppe interessiert ist, soll sich bei Uwe (251) melden. Bis jetzt existiert nur die Idee, aber die Frankfurter haben bewiesen, daß es geht ! Alles weitere am Telefon von Uwe. Also: WER macht mit ????????

- Wie schon im letzten Info ausführlich dargestellt, ist und bleibt unsere Finanzsituation alles andere als rosig. Außerdem sind die Druckpreise seit langer Zeit gleich, so daß eine Erhöhung immer wahrscheinlicher wird. "Sonstiges" und "Hardware" bleibt wohl auf absehbare Zeit unter 15%-Anteil. Die Aufwandsentschädigungen sind mit weniger als 6 DM/std. mehr als unterbewertet. Bei 5oo neuen Mitgliedern pro Jahr ließe dies sich zwar noch - wenn man optimistisch ist - mit Aufnahmegebühren auffangen. Aber was uns die Post neuerdings zumutet, ist eine Zumutung: Seit der letzten Ausgabe wird die Annahme der Infos mit geleimten Rücken als Büchersendung verweigert, obwohl die Sendungen gemäß dem Postbuch 8o als Büchersendungen zu werten sind. Das entspricht einer Erhöhung der Gebühren um 100 %. Am 1.1.82 erfolgen dann Erhöhungen auf breiter Front (Drucksache um 50%, Päckchen um ca. 40%, Brief um 33%). Das entspricht einer Erhöhung der Infoversandkosten (Porto) bezogen auf Büchersendungen um 200 % !!!! Dieses Finanzfiasko verdeutlicht nebenstehende Grafik unserer Ausgaben: Die Portokosten würden weit über 50% der Clubausgaben ausmachen. Und das, wo der Club sowieso schon in den roten Zahlen rangiert !!! Wir haben überlegt, was man tun könnte und kamen zu dem Entschluß, daß, wenn der Club nicht eingehen soll, eine Erhöhung des Clubbeitrages auf 60,--DM pro Jahr unvermeidbar wird. Dieser Beschluß fällt uns insbesondere deswegen schwer, weil ein Großteil (die Mehrheit) der Mitglieder Schüler und Studenten sind. Deren Einkommen richtet sich mehr oder weniger nach den Bafög-Sätzen und die werden 1982 nicht etwa wie die Postgebühren des Staates drastisch erhöht, sondern GESENKT !! Hier soll sich jedes Mitglied seinen eigenen Reim drauf machen, denn wir wollen in prisma keine Politik machen; nur soviel: Die unvertretbare Wirtschaftspolitik mit einer Schuldenpolitik, die den Steuerzahler in Kürze noch mehr abverlangen muß, weil die Zinsen die Neuverschuldung sogar schon übersteigen, kostet uns alle jetzt einen Tribut: Es ist sicher übertrieben, die Beitragserhöhung unseres Club nur der Politik der Bundesregierung zuzuschreiben, denn bisher waren wir immer deutlich in den Miesen, aber die SPD-Regierung wirkt über die Post auch bei unserem Club als größter Inflationstreiber: Auf die 40,--DM Jahresbeitrag aus 1981 kommen ca. 12 bis 14 DM für Postgebührenerhöhungen. Damit bleiben real nur etwa 6 bis 8 DM Mehreinnahmen für den Club, die, wie eifrige Clubmitglieder aus der Ausgabenstatistik der letzten Ausgabe errechneten, immer noch nicht geeignet sind, die Kosten auch nur annähernd zu decken. Wir sind bei einem Jahresbeitrag von 60,--DM also nach wie vor auf hunderte neuer Mitglieder 1982 (Aufnahmegebühren !) und wenig Austritte zum Jahresende 81 angewiesen ! Freilich bleibt auch noch eine andere Alternative: Einschränkung der Clubleistungen (Infos um ca. 50% dezimieren) und Beibehalten des Beitrag von 40 DM/J. Das würde uns wahrscheinlich aber ruinieren. Dennoch: Mit dem Wahlbogen 81 könnt Ihr alle Eure Meinung äußern: Beitrag 4o oder 6o DM pro Jahr ??????? Eine andere Alternativ existiert leider nicht, doch was sollen wir machen ? 60 DM/J. werden niemanden ruinieren und es geht ob en nicht anders .....

Diese Ausgabe umfasste 86 Seiten. Aus quantitativen Gründen mußten 25 davon für nächsten Monat zurückgestellt werden !

HEWLETT-PACKARD ANWENDER-CLUB

Ausgaben Januar bis Mai 1981

Drucke 46.6%

Sonstiges 8.3%

Hardware 4.1%

Honorare 23%

Porto 17.8%

grafik erstellt von klaus böttner, hewlett-packard gmbh, auf einem hp-85 mit plotter
danke, klaus !!! 100% = 16.594,24 dm

C L U B B Ö R S E

Clubmitglieder können in der Clubbörse kostenlos Gelegenheitsanzeigen aufgeben.

Verkaufe HP-41c (09/79, Nr. 1940...), 2 DDM, Kartenleser (12/80), HP-Akku-/Netzgerät, Buch "SYnthetic Prgr." von Dr. Wickes für 720 DM, u.U. auch einzeln.
Johannes Schu, Ludweilerstr. 181, 6620 Völklingen 7 (129)

Verkaufe HP-19c, 5 Monate Garantie, komplett und 1a Zustand. Preis: VHB 450 DM
Karsten Peierlein, Breierspfad 83, 4600 Dortmund - Wambel (093)

Suche zwei Batteriehalter für HP-41
Heiner Krause, Hellmundstr. 4, 6200 Wiesbaden 1 (473)

Verkaufe PET 2001, 8K, kleine Tastatur mit Datasette, einwandfreier Zustand, 900 DM
sowie EPSON-Drucker MX-80 mit cbm-Interface für Groß- und Kleinschreibung und Umlaute, neu gekauft Januar 1981 (2069 DM) für 1200 DM
Beide Geräte zusammen 2000 DM.
Prof. Dipl. Ing. H.L. Jochen Kux, Pommernstr. 90, 6800 Mannheim 31 (452)

Verkaufe PPC-ROM nagelneu mit zwei Handbüchern und allem Zubehör
Einmalige Gelegenheit, da nicht mehr produziert wird !
Festpreis 300,--DM plus 15,--DM Porto und Verpackung
Oliver Rietschel, Postfach, 2420 Eutin 1 (001)

Angebot: 1) Lieferung eines Zusatzgerätes zum Netzteil des 41-Printers, so daß Rechner über Netz mit Strom versorgt wird und 1 Satz Akkus Typ Varta 151 D außerhalb geladen wird.
2) wie 1) jedoch mit Netzteil für Nicht-Drucker-Besitzer, wobei späterer Anschluß am Printerbetzteil möglich ist.
3) Varta Akkus Typ 151 D für je 4,80 DM (zuzügl. Porto 3,--)

zu 1) und 2): Lieferumfang: Gerät (geprüft/getestet), Beschreibung mit Anbauanleitung sowie wichtigen Bedienungsanweisungen für den Betrieb
Lieferzeit: ca. 1 Woche
Garantie: Gerät getestet und geprüft. Keine weitere Garantie
Preis: Gerät wie Pkt. 1: 30,--DM
Gerät wie Pkt. 2: 45,--DM
Porto je Bestel.: 3,--DM
Zahlung: Scheck im vorab

Gerhard Schönemann, Kirchfeld 11, 3584 Zwesten, Tel. 05626-1458 (472)

L E S E R B R I E F E

Grundlagen !
Synthetische Programmierung ist nicht nur etwas für Byte-Dompteure sondern kann durchaus helfen, Programme zu verkürzen, oder macht gewisse Routinen erst möglich. Für alle diejenigen, denen nur an der praktischen Anwendung der synthetischen Programmierung gelegen ist, und die vielleicht weder Muße noch Gelegenheit haben, sich durch den Wickes oder das PPCJ zu wühlen, sollte unter Federführung eines "Synthetic-Profi" eine Rubrik in "prisma" errichtet werden, wo in Form einer Fortsetzungsserie die wichtigsten Befehle der synthetischen Programmierung nach der Devise abgehandelt werden:
1. Was bewirkt der Befehl ?
2. Für welche Zwecke kann man ihn anwenden ?
3. Wie kann man den Befehl erzeugen ?
Hintergrundinformationen über Byte-Dressurakte sollten dem Leser von Wickes und natürlich weiterhin speziellen Artikeln in prisma vorbehalten bleiben.
Klaus Werner Hoenow (11) Harald Wienbeck (141)

Hallo Clubmitglieder ! Wer forscht mit ?
In unserem HP-41 stecken noch viele Geheimnisse !!! Als Forschungsgrundlage dient das Programm "LOAD BYTES"; zu beziehen durch den Kartencopyservice. Das Prgr. wird wie

folgt bedient: 1. Karten mit LB einlesen. GTO..
2. Eingeben: 01 LBL"AB" 02 XEQ"LB" 03 STOP 04 ENTER 05 ENTER .....
Soviel ENTER eingeben, wie benötigt werden, besser zuviel als zuwenig
3. XEQ"AB" Das Prgr. Load Bytes wird nun automatisch aufgerufen und fügt die Programmbefehle anstelle der ENTER ein. Es müssen nacheinander die Prgr.Befehle in dezimalem Code eingegeben werden (prisma 27/28 - 1980).

Beispiel: Du willst STO M RCL M X()IND e GTO IND M erzeugen.
Dazu ist dezimal folgendes einzugeben: 145,117/144,117/206,254/174,117/

Es können auch Alphalabel und ENDmarken erzeugt werden. Dezimaler Code der Alphalabel ist 192, 00 (=Tastenzuordnung), 242 (=TEXT2), 66 (=A), 67 (=B)
Wenn zu wenig ENTER eingegeben wurden, erfolgt die Fehlermeldung NO MORE. Wenn alle Programmbefehle in dezimalem Code eingegeben sind, R/S drücken, ohne einen Wert einzugeben. Beim Umschalten in den Programmodus siehst Du evtl. nach einigen ENTER die erzeugten Programmbefehle. Nach Löschen der überflüssigen ENTER steht das Programm zur Verfügung. Mit diesem Prgr. sind alle möglichen Befehle zu erzeugen, u.a. auch Befehle, die völlig neue Wirkungen zeigen. Z.B. belegt die ENDmarke (193, 00, 00) sämtlichen vorhandenen Speicherplatz und ist nur durch MEMORY LOST zu löschen. Bitte experimentiert doch einmal und versucht u.a. herauszufinden, was mit den versch. LBL und ENDmarken möglich ist. Euer Walter (205)

Teuerung bei der Post

Gebührenanstieg Ende 1971 bis 1982 (geplant)

	1971	72	74	79	1982
Drucksache	20	25	30	40	60
Postkarte	25	30	40	50	70
Brief (in Pf)	30	40	50	60	80
Päckchen (in DM)	1,20	1,50	2,00	2,30	3,20
Paket (1. Zone, bis 5 kg, in DM)	1,80	2,20	3,10	3,40	4,50

Die Post greift ihren Kunden in die Tasche, weil der Bund der Post in die Tasche greift. So begründet die Post ihre für März 1982 geplanten Gebührenerhöhungen. Denn der Bund als Eigentümer der Post will statt bisher 6⅔ Prozent künftig 10 Prozent der Posteinnahmen in seine Kasse leiten. Dieser Aderlaß würde — so die Post — dazu führen, daß die Gewinne rapide schrumpfen und schon 1983 in Verluste umschlagen. Um das zu verhindern, sollen die Gebühren vor allem in jenen Bereichen angehoben werden, in denen die Erträge nicht die Kosten decken. Und das ist nahezu überall der Fall, wenn man einmal vom gewinnträchtigen Fernmeldedienst absieht.

Aus: Sylter Nachrichten

E P R O M s für H P - 41

In der letzten prisma-Ausgabe wurde zum erstenmal ein Kassetteninterface für den HP-41 vorgestellt. Diese Ausgabe folgt nun mit der "Barcodeplottsensation", einem Superknüller. Beide Sachen wurden nirgendwo sonst in der Welt publiziert, auch nicht in den USA ! Dafür verlautete aber etwas anderes über den großen Teich. Eine private amerikanische Firma bietet ab etwa Ende August/Anfang September Eproms für den HP-41 an. Ein EPROM ist in der Anwendung einem ROM gleichzusetzen, hat jedoch den Vorteil, mit einem entsprechendem Gerät (Programmer) vom Benutzer beliebig oft selbst programmiert werden zu können. Was bringt nun diese Neuerung für den 41 ???

Vorteile:

+ billiger als Kleinserien von ROMs: n ROMs kosten 2o ooo + n · 22 US $
  n EPROMs kosten 1 ooo + n · 250 US $

  Die EPROM-Preise sind Schätzwerte, wie sie in den USA der Presse mitgeteilt wurden. Dieses bedeutet: Ab 84 Modulen sind die EPROMS teurer, darunter aber billiger als die ROMs. Dabei wurde für die EPROMS ein Programmer und Materialkosten excl. Programmierkosten veranschlagt. In der Praxis werden sich die EPROMs evtl. noch bei steigender Nachfrage verbilligen, hinzu kommt aber noch der Arbeitslohn etc. fürs Programmieren, wenn man dies nicht rein aus Hobby tut.

+ EPROM-Inhalte können nachträglich geändert werden (Update-Versionen !)

+ Extrem viel schnellere Programmausführung bei Programmierung in der HP-41 eigenen Maschinensprache (nicht UPN !), die nur über EPROMs anzusprechen ist (derzeit), im Vergleich zur UPN-Programmierung.

+ Es besteht die Möglichkeit, völlig neuartige Aufgaben zu bewältigen und dabei noch Speicherplatz einzusparen im Vergl zur UPN-Programmierung. Beispiele wären u.a. z.B.: höhere Programmiersprachen, wie BASIC, FORTRAN, APL, ..., völlig neue Alphazeichenorganisation

+ Mit der über EPROMs zugänglichen Maschinensprache ist es möglich, Befehle zu definieren, die im Zusammenhang mit der von uns veröffentlichten "Strichcodeprintingmethode" auf dem 41-Standarddrucker sensationelle Anwendungen ermöglichen. Ich denke da z.B. an einen Befehl, der z.B. PRB heißen könnte und ähnlich wie PRP wirkt, nur mit dem Unterschied, daß das Programm nicht als gewöhnliches Listing, sondern als fertiger Strichcode auf dem Drucker ausgegeben wird. Eingabewert wäre dann - ähnlich PRP - nur das Label des zu druckenden Programms ! Das erscheint mir vorerst zwar noch Zukunftsmusik, aber die Zukunft kommt ja schließlich .....

Nachteile:

- Sehr teuer: Vergleich mit ROMS siehe oben. Das erste EPROM kostet mit Programmer 125o US $ (nicht unter 35oo,--DM mit Exportkosten). Jedes weitere EPROM kostet den Anwender dann 250 US $ (ca. 7oo,--DM). Eine größerer Absatz von EPROMS dürfte also noch eine Preissenkung voraussetzen, die aber möglich zu sein scheint. Vergleich mit RAMs: Einfacher und billiger ist das sogenannte Page-Switching, also der Einbau mehrerer QDMs in den Rechner, wie es Hans-Günther macht. Diese Methode hat natürlich auch ihre Vor- und Nachteile.

- Eliminierung jeglichen Kopierschutzes. Selbst PRIVATE-geschützte ROMs können mit EPROMs - wenn man die Maschinensprache kennt - problemlos entprivatisiert und/oder kopiert werden.

- Der eigentliche Vorteil - extreme Vervielfachung der Rechengeschwindigkeit - ist derzeit unrealistisch. Die Programmierung in Maschinensprache wird von HP aus verständlichen Gründen ebenso wie die synthetischen Programmierungen nicht unterstützt; es ist sogar mehr als verständlich, wenn man sich vor Augen hält, daß zur effektiven Programmierung nur hochqualifizierte Fachleute in der Lage sein werden, deren Arbeitslohn die EPROMs dann wohl dermaßen verteuern wird, daß gute EPROMs wahrlich unbezahlbar werden. Die "bezahlbaren" Eproms werden sich daher wohl vorerst auf "normale"Programme beschränken, die dann aber auch nicht wesentlich schneller als im RAM oder ROM laufen....

Die genannten 250 US $ beinhalten ein EPROM und ein EPROM-Interface. Werden mehrere EPROMs wahlweise an einem Rechner benötigt, so ist nur ein Interface nötig.

Neue Projektleiter ! Oliver (1)

Wie immor, wenn es etwas zu tun gibt, ist die Resonanz auch bei den Projektleitergesuchen sehr mäßig gewesen. Darum ein paar Klärende Worte:
Ein Projektleiter ist für das von ihm selbst gewählte Themengebiet als Koordinator tätig. Bei Softwarethemen sieht das in etwa so aus:
Beispiel Mathematik: Alles, was bisher an mich gesandt wurde und zum Thema Mathematik paßt, geht jetzt an den zuständigen Projektleiter. In diesem Fall (Mathematik) ist das z.B. vom Plottprogramm für mehrere Funktionen bis zu Integralrechnungsprogrammen alles was so anfällt. Auch einfache und einfachste Sachen sind erwünscht, beispielsweise die Sortierroutinen in dieser Ausgabe oder auch die (hoffentlich sehr vielen !!!) BUBBLE-SORTs, die zu schreiben in dieser Ausgabe aufgerufen wird. Eben alles, was mit Mathematik zu tun hat.
Der Projektleiter ordnet dann die erhaltenen Programme ähnlich, wie ich es mit den SORT3-Programmen getan habe und schickt seinen Aufsatz an mich zum Druck. Bei größeren Programmen hat er darauf zu achten, daß nichts doppelt oder in sehr ähnlicher Form zweimal veröffentlicht wird. Hierbei hat er zwischen doppelten und sich ergänzenden Programmen zu unterscheiden. Werden Programme deutlich verbessert, so sind diese auch gerne für prisma erwünscht; selbst dann, wenn sie in der Funktion mit vorgehend veröffentlichten identisch sind.
Für die Mitglieder bedeutet ein neuer Projektleiter nur eine neue Adresse, an die sie Ihre Prgr. schicken möchten - und für den Projektleiter oft einen erheblichen Nutzen für den eigenen Wissensstand oder die eigene Ausbildung !
Für alle - ob Projektleiter oder nicht - gilt natürlich nach wie vor: Platz in prisma bestmöglich ausnutzen ! Darum: Schreibmaschine einzeilig und keinen übermäßig großen Rand (insbesondere nicht in der Mitte zwischen den beiden Textspalten !!!) Vergleicht doch mal die von mir geschriebenen Seiten mit denen von Niels oder den vielen anderen, die immer noch eineinhalb- bis zweizeilig schreiben. Übersichtlichkeit: JA aber: PREISGÜNSTIGER DRUCK: UNBEDINGT !!
Damit wünsche ich uns allen einen schönen Spätsommer, der auch hoffentlich ein paar neue Überaschungen für den Club bringt ! Euer Oliver

---

PROJEKTLEITER MATHEMATIK

ANDREAS WOLPERS - KRAMERSTR. 1 - 3200 HILDESHEIM - NR. 349
NR. 349

PROJEKTLEITER ELEKTROTECHNIK

MARIUS HEYN - GROSSHAUSBERG 02 - 10 - 03 - 7743 FURTWANGEN - NR. 164

Günter Odebrecht, Am Stadtpfad 11, 6236 Eschborn, Tel. o6196-46238 würde evtl. eine Sammelbestellung Batterien für den 41 zum sehr günstigen Preis durchführen. Schreibt ihm bitte kurz auf einer Postkarte, wie groß Euer Interesse an einer solchen Sammelbestellung (Preis ca. 1,6oDM / Batterie) ist.

Umwandlung von Dezimalzahlen in Hexadezimalzahlen: Das Prgr. verarbeitet Zahlen bis zu über 9.999.999.999, enthält keine synthetischen Befehle und benötigt keinen Datenspeicher. Johannes Schu (129)

```
LBL"DH"   CLA    LBL 99   STO Y    16    MOD    ASTO T    ASHF    ASTO Z
XEQ IND X    ARCL T    ARCL Z    X()Y    LASTX    /    INT    X≠0?    GTO 99
PROMPT
LBL 00    "0"    RTN    LBL 01    "1"    RTN ........... LBL 15    "F"    RTN
END    102 Bytes
```

In der letzten prisma-Ausgabe erschien das SORT3 Programm von Guido Studer. Da es zu verkürzen war, nicht immer richtig funktionierte und schließlich einen Großteil der Clubmitglieder interessierte, erhielt dieses Programm die größte Resonanz, die je eines seit Bestehen des Clubs erreicht hat ! Ca. 50 versch. verbesserte SORT3 s gingen bei mir ein, von denen einige hier vorgestellt werden sollen.
Nebenstehende Version ist weder kurz (36 Bytes), noch sonderlich schnell, dafür aber recht originell. Außerdem hat Michael noch ein Bubble Sort Prgr. geschrieben, wer schickt ein verbessertes Bubble Sort (oder sollte das schon optimal sein ?) ?

Kürzer ist da schon die Version von Ulli(3), der es auf 30 Bytes brachte. Doch in wenigen Sekunden war auch nebenstehende Routine von Michael (421) auf 30 Bytes verkürzt; doch dann kam noch ein Michael: Michael (325) sandte mir eine 28-Byte-Version, die die beiden anderen unterbot. Er sortierte nur im STAK und sparte dadurch Speicherbefehle ein:

```
01♦LBL "SORT3"
RCL 02  RCL 01  X>Y?
X<> 02  STO 01  RCL 03
X<=Y?  X<> 01  X<> 02
STO 03  RCL 02  X>Y?
X<> 03  STO 02  END
                    CAT 1
LBL'SORT3
END          30 BYTES
             ULLI -3-
```

```
01♦LBL "SORT3"
RCL 03  RCL 02  RCL 01
X>Y?  X<>Y  RDN  X>Y?
X<>Y  R↑  X>Y?  X<>Y
STO 01  RDN  STO 02  RDN
STO 03  END
                    CAT 1
LBL'SORT3
END          28 BYTES
             MICHAEL -325-
```

Andreas (69) und Gerhard (197) unterboten aber selbst diesen Rekord.
Ihre beiden Versionen benötigen nur 26 Bytes ! Diese beiden Versionen sind denn auch die kürzesten aller mir zugesandten !

```
01♦LBL "SORT3"
RCL 02  RCL 03  X<Y?
X<>Y  RCL 01  X<Y?  X<>Y
STO 03  RDN  X<Y?  X<>Y
STO 02  X<>Y  STO 01
END
                    CAT 1
LBL'SORT3
END          26 BYTES
             ANDREAS -69-
```

```
01♦LBL "SORT3"
RCL 01  RCL 02  X>Y?
X<>Y  RCL 03  X>Y?  X<>Y
STO 01  RDN  X>Y?  X<>Y
STO 02  RDN  STO 03  END
                    CAT 1
LBL'SORT3
END          26 BYTES
             GEORG -197-
```

Doch halt: 2 Schummler waren auch noch darunter ! Gerhard (472) und Albert (278) schrieben <u>unabhängig</u> voneinander das nebenstehende Programm, das nur 20 Bytes Speicherplatz benötigt.

```
01♦LBL "SORT3"
X>Y?  X<>Y  RDN  X>Y?
X<>Y  R↑  X>Y?  X<>Y
END
                    CAT 1
LBL'SORT3
END          20 BYTES
             GERHARD -472-
             ALBERT -278-
```

Aber warum Schummler !?? Nun, diese Routine sortiert nur im Stack und beachtet damit nicht die Vorraussetzungen, die die Aufgabe stellte ! Aber selbst, wenn man drei RCLs vor die Routine hängt, und die Ergebnisse auch noch abspeichert, braucht man nur 28 Bytes. Und beim Vergleich mit dem Prgr. von Michael (325) sieht man ja auch, daß ......

Man merkt: Diese Aufgabe machte vielen Mitgliedern Spaß, darum auf in die nächste Runde: Wer schreibt das kürzeste BUBBLE-SORT-Prgr. ??? WICHTIG: Die Programme keinesfalls an Oliver (1) schicken, sondern an den neuen Projektleiter Mathematik !!!!

Und so sieht es aus, wenn ein ROM im HP-41 fehlerhaft ist (von Georg (197)):

```
CAT 1
26 BYTES
.END.
86 BYTES
CLX
0.0000
CAT 1
26 BYTES
.END.
86 BYTES
CAT
```

SORTIEREN

a)
Zuerst möchte ich auf die beiden Programme von Guido Studer (184-81) eingehen : Meiner Meinung nach enthalten beide mehrere Fehler. Abgesehen von der Tatsache, daß in meiner Kopie beim Basic Programm die logischen Vergleichssymbole fehlen, ist es überhaupt nicht möglich, mit zwei Vergleichsbefehlen drei Zahlen zu sortieren. So liefert die 41er Version bei fast allen Eingaben falsche Ergebnisse. Es wird z.B. 3(R01) 1(R02) 2(R03) zu 1(R01) 3(R02) 2(R03) "sortiert"
Hier mein eigenes Programm, das nur deshalb mit zwei Vergleichsbefehlen auskommt, weil in Zeile 014 ein Rücksprung vorgesehen ist :

```
001 LBL "SORT3"
    RCL 01
    RCL 02
    X>Y?
    GTO 01
    X<>01
    STO 02
008 LBL 01
    RCL 03
    X>Y?
    GTO 02
    X<>02
    STO 03
    GTO "SORT3"
015 LBL 02
016 END
```

VERKUERZT:

```
01•LBL "SORT3"
02•LBL 02
RCL 01  RCL 02  X>Y?
GTO 01  X<> 01  STO 02

09•LBL 01
RCL 03  X>Y?  RTN
X<> 02  STO 03  GTO 02
END
                CAT 1
LBL'SORT3
END         38 BYTES
```

b) BUBBLE SORT

Hier nun ein Programm, mit dem beliebig viele Zahlen, die in beliebigen hintereinanderliegenden Registern stehen, der Größe nach sortiert werden können.
Nach Eingabe von REG a (Anfangsregister) und REG e (Endregister) werden die zwischen diesen beiden Schranken liegenden Inhalte der Register sortiert. (REG a > 04)
Die verwendete Methode nennt sich Bubble Sort (vergl. CHIP 7/81 S.46ff)

```
001 LBL "BSORT"
    "REG a ?"
    PROMPT
    STO 03
    "REG e ?"
    PROMPT
    STO 04
008 LBL 01
    RCL 03
    STO 01
    1
    +
    STO 02
014 LBL 02
    RCL IND 02
    RCL IND 01
    X<=Y?
    GTO 03
    X<>Y
    SF 00
021 LBL 03
    STO IND 01
    RDN
    STO IND 02
    1
    ST + 01
    ST + 02
    RCL 01
    RCL 04
    X>Y?
    GTO 02
    FS?C 00
    GTO 01
034 END
```

Viel Spaß beim Ausprobieren und Happy Programming

wünscht Euch

DIFFERENTIALGLEICHUNGEN

AUFLOESUNG VON ..

1. ORDNUNG:

Y'=F(X,Y)

2. ORDNUNG:

Y''=F(X,Y,Y') ODER
Y'=F(X,Y,Z)
Z'=G(X,Y,Z)

4. ORDNUNG:

Y''''=F(X,Y,Y',Y'',Y''') ODER
Y''=F(X,Y,Y',Z,Z')
Z''=G(X,Y,Y',Z,Z')

ABLAUF:

FUER PLOTTEN PRINTEN:
PRINTERSCHALTER: MAN
PLOTTEN: XEQ PRPLOT
  NAME?: DIFFGL ...
MIT PLOT2: USER-MODE
PRINTEN U/O ANZEIGEN:
  XEQ DIFFGL
F() RESP. G() ALS EIGENE
PROGRAMME SCHREIBEN
X STEHT IM X-REG. F()
RESP. G() MUESSEN AUCH
IM X-REG STEHEN.

STATUS:

R00-R10,R12-R19
ZUSAETZLICH BEI TYP 2:
R20-R24
ZUSAETZLICH BEI TYP 4
R25-R30
F01,F02,F04,F10,F19-F21

INHALT:

R08 X0,R09 XMAX,R10 XINC
R12 Y0,R13:"F()",R14 Y
R15:k,R16:kI,R17:I,R18 e
R19:n,R20:Z0,R21:"G()"
R22:Z,R23:1,R24:lI
R25:Y'0,R26 Z'0,R27:Y'
R28:Z',R29:k',R30:l'
F01-F02-F04 TYP,F10:BEL.
F19 AUX,F20:PLOT,F21 BEL

VERFAHREN: RUNGE-KUTTA
AUSGABE: 1. ANZEIGEN
         2. PRINTEN
         3. PLOTTEN
VOM PROGRAMM HER ODER
EXTERN: AUXILIARY
LITERATUR: LOESUNG VON
DIFFERENTIALGLEICHUNGEN
MIT PROGRAMMIERBAREN
TASCHENRECHNERN
AUTOR: G. WENZ
VERLAG: OLDENBOURG
BEARBEITET VON
MAX HUBER (1986)

```
01+LBL "DIF
       FGL"
02 FC? 19
03 FS? 20
04 GTO 15
05 XEQ 17
06 "PLOT:SF
       20"
07 PROMPT
08 "AUX:SF1
        9"
09 PROMPT
10 "TYP=1,2
       ,4?"
11 PROMPT
12 SF IND X
13 FS? 20
14 GTO 00
15 "XMAX?"
16 PROMPT
17 STO 09
18 "XINC?"
19 PROMPT
20 STO 10
21 "e?"
22 PROMPT
23 STO 18
24 CLX
25 STO 19

26+LBL 00
27 AON
28 "NAME:F(
       )?"
29 PROMPT
30 ASTO 13
31 FS? 01
32 GTO 05
33 "NAME:G(
       )?"
34 PROMPT
35 ASTO 21

36+LBL 05
37 AOFF
38 "X0↑Y0"
39 FC? 01
40 "⊢↑Z0"
41 "⊢?"
42 PROMPT
43 FC? 01
44 STO 20
45 FC? 01
46 RDN
47 STO 12
48 X<>Y
49 STO 08
50 FC? 04
51 GTO 19
52 "Y'0↑Z'0
       ?"
```

EINGABE SCHON GEMACHT? WEITERRECHNEN. F19,F20 müssen zum gelöscht sein.

EINGABE DER WERTE. ↓

BEI PLOT. UEBERSPRINGEN, DA SCHON ABGEFRAGT ODER NICHT BENOETIGT. e ist Genauigkeit für die Ausgabe (Stellen)

ANFANGSBEDINGUNGEN.

```
53 PROMPT
54 STO 26
55 X<>Y
56 STO 25

57+LBL 19
58 FS? 19
59 RTN
60 FC? 20
61 GTO 06
62 RCL 12
63 FS? 01
64 RTN
65 RCL 20
66 XEQ "PLO
       T2"
67 RTN

68+LBL 06
69 "X"
70 8
71 XEQ 16
72 "Y"
73 12
74 XEQ 16
75 FS? 01
76 GTO 15
77 "Z"
78 20
79 XEQ 16
80 FS? 02
81 GTO 15
82 "Y'"
83 25
84 XEQ 16
85 "Z'"
86 26
87 XEQ 16

88+LBL 15
89 RCL 08
90 RCL 09
91 X<=Y?
92 GTO 17
93 CLX
94 STO 15
95 FC? 01
96 STO 23
97 FS? 04
98 STO 29
99 FS? 04
100 STO 30
101 2
102 ST/ 10
103 X↑2
104 STO 17

105+LBL 18
106 XEQ IND
        17
107 RCL 08
```

AUSGABE: AUX? PLOT MIT PLOT2? ODER PRINTEN?

BERECHNUNG VON k, l, k', l'

```
108 XEQ IND
         13
109 STO 16
110 ST+ 15
111 FS? 10
112 ST+ 15
113 FS? 01
114 GTO 07
115 FS? 04
116 ST+ 29
117 RCL 08
118 XEQ IND
         21
119 STO 24
120 ST+ 23
121 FS? 10
122 ST+ 23
123 FS? 04
124 ST+ 30

125•LBL 07
126 DSE 17
127 GTO 18
128 FC? 04
129 GTO 08
130 ST- 30
131 RCL 16
132 ST- 29
133 RCL 10
134 6
135 /
136 ST* 29
137 ST* 30

138•LBL 08
139 RCL 10
140 6
141 /
142 ST* 15
143 FC? 01
144 ST* 23
------------------------------
145 ISG 19
146 CLX
147 FS? 04
148 GTO 11
149 RCL 15
150 ST+ 12
151 FS? 01
152 GTO 19
153 RCL 23
154 ST+ 20
155 GTO 19

156•LBL 11
157 RCL 29
158 RCL 25
159 +
160 RCL 10
161 *
162 ST+ 12
163 RCL 15
164 ST+ 25
165 RCL 30
166 RCL 26
167 +
168 RCL 10
169 *
170 ST+ 20
171 RCL 23
172 ST+ 26
173 GTO 19
------------------------------
NAECHSTE PUNKTE.

174•LBL 16
175 FIX 0
176 ARCL 19
177 "⊢= "
178 RCL 18
179 LOG
180 1
181 +
182 FIX IND
          X
183 ARCL IND
          Y
184 AVIEW
185 RTN
------------------------------
U-PROGRAMM DER AUSGABE.

186•LBL 01
187 CF 10
188 RCL 10
189 2
190 ST* 10
191 X<>Y
192 GTO 09

193•LBL 03
194 SF 10
195 RCL 10

196•LBL 09
197 ST+ 08

198•LBL 02
199 XEQ 04
200 RCL 16
201 FS? 04
202 GTO 10
203 ST+ 14
204 FS? 01
205 RTN
206 RCL 24
207 ST+ 22
208 RTN

209•LBL 10
210 RCL 25
211 X<>Y
212 +
213 STO 27
214 LASTX
215 2
216 /
217 -
218 RCL 10
219 *
220 ST+ 14
221 RCL 26
222 RCL 24
223 +
224 STO 28
225 LASTX
226 2
227 /
228 -
229 RCL 10
230 *
231 ST+ 22
232 RTN

233•LBL 04
234 RCL 12
235 STO 14
236 RCL 10
237 ST* 16
238 FS? 01
239 RTN
240 ST* 24
241 RCL 20
242 STO 22
243 FS? 02
244 RTN
245 RCL 25
246 STO 27
247 RCL 26
248 STO 28
249 RTN
------------------------------
UNTERPROGRAMM FUER s1.

250•LBL 17
251 CF 01
252 CF 02
253 CF 04
254 CF 10
255 CF 19
256 CF 20
257 BEEP
258 "FIN"
259 CF 21
260 AVIEW
261 SF 21
262 END

LBL'DIFFGL
END
529 BYTES
```

Programmierung von Peripheriefunktionen ohne Anschluß der Peripherie-Einheiten.

Obwohl ich sowohl den Kartenleser als auch den Drukker besitze, schließe ich die Geräte nur an den Rechner an, wenn ich sie tatsächlich benötige.

Inzwischen besitze ich zwar das Quad-Modul und könnte daher gleichzeitig beide Geräte angeschlossen lassen, aber vorher, bei der Verwendung dreier Single-Module, hatte ich nur die Möglichkeit, beide Geräte abwechselnd anzuschließen.

Ich empfand es schon immer als sehr lästig, die Geräte anschließen zu müssen, wenn ich deren Funktionen in ein Programm übernehmen wollte.

Mehr durch Zufall,als daß ich danach gesucht habe, fand ich heraus, wie es möglich ist, die Funktionen zu programmieren,ohne die entsprechenden Geräte anschließen zu müssen.

Nach meinem Wissensstand gibt es 2 Hilfsmittel.

1.) Den Byte-Jumper

2.) Das Byte-Loading-Programm

In diesem Fall verwende ich lieber den Byte-Jumper, weil er nach meiner Meinung weniger Aufwand erfordert.

Im PPC Calculator Journal V8N1P31 wird der

Program Mode "Byte-Jumper"

beschrieben. Dieser "F3-Byte-Jumper" hat gegenüber dem Standard F1-Byte-Jumper den Vorteil, daß er sowohl im Run-Modus als auch im Pgrm-Modus wirkt.

Für die Hilfe des F3-Byte-Jumpers beim Programmieren von XROM-Funktionen möchte ich folgendes Beispiel bringen.

Nehmen wir einmal an, wir möchten die Funktion PRREG programmieren. Wir bleiben im Pgrm-Modus und geben folgende Funktionen ein.

```
1 E        (EEX-Taste)
STO IND 39
LN
```

Jetzt gehen wir zurück zur Zeile 1 und löschen den Befehl mit der "Backarrow"-Taste. Wir bleiben weiter im Pgrm-Modus, verändern weiter nichts und setzen jetzt lediglich den F3-Byte-Jumper ein.

Wir sehen jetzt das hochgestellte "T" für Text, den Querstrich am oberen Display-Rand als Display-Ausdruck für das "Null"-Byte und zweimal den "Boxed Star".

Diese Zeile können wir jetzt wieder mit "Backarrow" löschen und betätigen einmal SST. In der Zeile, die wir sehen, steht XROM 29,16. Wenn wir den Drucker anschließen, wird die Funktion ganz richtig als PRREG interpretiert.

In dieser Weise kann man jede Funktion des Druckers und des Kartenlesers programmieren. Nur die Funktionen des Bar-Code-Lesers kann man so nicht schaffen. Das geht tatsächlich nur mit dem Byte-Loading-Programm.

Warum verwenden wir bei der "synthetischen" Programmierung von XROM-Funktionen gerade die Befehle, wie sie im Beispiel oben verwendet wurden?

In Zeile 1 steht ein "2-Byte"-Befehl (1 E). Jeder 2-Byte-Befehl kann verwendet werden. "1 E" ist nur deshalb so praktisch, weil nur einmal die EEX-Taste gedrückt werden braucht.

Diesen 2-Byte-Befehl tauschen wir durch den 3-Byte-Befehl "F3-Byte-Jumper" aus. In der nächsten Zeile steht wieder ein 2-Byte-Befehl ((STO) (IND 39)).

Durch den Byte-Jumper wird Byte 1 (STO)"geschluckt". Byte 2 (IND 39) wird nun vom Rechner auf einzig mögliche andere Weise interpretiert. Nämlich als XROM. Der 1-Byte-Befehl "LN" wird damit automatisch als "PRREG" verstanden.

Bei der Programmierung von Funktionen des Kartenlesers und des Druckers muß in Zeile 2 immer der Befehl "STO IND 39" verwendet werden. Nur der Befehl in Zeile 3 wechselt, je nach dem, welche Funktion wir programmieren möchten.

Hier eine Liste der Befehle, die der gewünschten Funktion entsprechen. Die Zahlen in der Liste sind die Eingabezahlen bei Verwendung des "Byte-Loading-Programms".

MRG	RAD	167,129
RDTA	GRAD	167,130
RDTAX	ENTER	167,131
RSUB	STOP	167,132
VER	RTN	167,133
WALL	BEEP	167,134
WDTA	CLA	167,135
WDTAX	ASHF	167,136
WPRV	PSE	167,137
WSTS	CLRG	167,138
ACA	-	167,065
ACCHR	x	167,066
ACCOL	/	167,067
ACSPEC	X(Y?	167,068
ACX	X)Y?	167,069
BLDSPEC	X(=Y?	167,070
LIST	Σ+	167,071
PRA	Σ-	167,072
PRAXIS	HMS+	167,073
PRBUF	HMS-	167,074
PRFLAGS	MOD	167,075
PRKEYS	%	167,076
PRP	%CH	167,077
PRPLOT	P-R	167,078
PRPLOTP	R-P	167,079
PRREG	LN	167,080
PRREGX	X hoch 2	167,081
PRΣ	SQRT	167,082
PRSTK	Y hoch Y	167,083
PRX	CHS	167,084
REGPLOT	E hoch X	167,085
SKPCHR	LOG	167,086
SKPCOL	1Ø hoch X	167,087
STKPLOT	E hoch X-1	167,088

Was sind der Byte-Jumper und das Byte-Loading-Programm? Inzwischen dürfte verständlich geworden sein, daß beides Hilfsmittel bei der "synthetischen" Programmierung sind. Das LB-Programm gibt uns die Möglichkeit, jedes gewünschte Byte in fast jeder gewünschten Anzahl pro Zeile in Programmzeilen hineinzuladen.

Der Byte-Jumper ist eine synthetische Funktion, geschaffen mit dem KA (Key Assignment) - Programm oder dem MK (Multiple Key Assignment) - Programm und einer von uns gewünschten Taste zugeordnet.

Im USER-Modus verwendet, veranlaßt er den Rechner, jedes von uns gewünschte Byte auf die einzig andere mögliche Art zu interpretieren.

Wie kommt man an den Byte-Jumper? Durch eine Status-Karte, das KA- oder das MK-Programm.

Die Status-Karte bekommt man vielleicht, das KA-Programm ganz sicher beim Karten-Copy-Service unseres Clubs oder übernimmt es aus dem Prisma Seite 151,152 Juni '81.

Für den Karten-Copy-Service verweise ich aufs Prisma Seite 145,146 Juni '81.

Inzwischen gibt es mehrere verschiedene Ausführungen des KA-Programms. Mir selber gefällt eines am besten, das auf 2 Magnetkarten gespeichert ist.

Für das MK-Programm sind 3 Karten nötig. Ich finde es zwar noch besser, es ist leider aber auch länger. Es ist die RAM-Version des Programms aus dem PPC-Custom-ROM.

Als Eingabevorschlag möchte ich die Zahlen 243 und ~~241~~ 142 weitergeben.

Happy Programming

Niels Nöhren (81)
Kielort 16
2000 Norderstedt
Telefon (040) 524 28 88
17.06.1981

Eine neue Variation der Clear-Routine.

In seinem Artikel über die Rechnerorganisation in der Prisma-Juni-Ausgabe auf den Seiten 147 bis 151 '81 beschreibt Andreas Marktscheffel, wie man den Adress-Pointer des Sigma-Register-Blocks im Register c auf "ØØØ" setzen kann.

Anschließend kann man mit dem Befehl "Clear-Sigma" den gesamten Stack einschließlich des L-Registers und dem Alpha-Register M löschen.

Andreas' Beschreibung hat mich zum Nachdenken angeregt und ich bin auf die Idee gekommen, den Sigma-Register-Block ganz grundsätzlich mal im Stack zu positionieren.

Wenn ich also den Block im Register T beginnen lasse, und wenn ich dann "Clear-Sigma" ausführe, erreiche ich genau das, was Andreas beschreibt.

Natürlich ist es nicht auf normalem Wege möglich, die Sigma-Register in den Stack zu bekommen, aber mit der Hilfe des Byte-Jumpers ist es einfach.

In der Clear-Routine habe ich die Sigma-Position in das Register L gebracht. Jetzt kann ich mit Clear-Sigma die Register L, M, N, O, P und Q löschen.

Meine neue Routine sieht jetzt so aus:

```
Ø1 LBL "CLR"
Ø2 ΣREG L
Ø3 CLΣ
Ø4 ΣREG Ø1
Ø5 CLST
Ø6 STO d
Ø7 SF 29
Ø8 FIX 2
Ø9 OFF
1Ø END
```

Mit dem Befehl in Zeile 4 bringe ich den Block wieder aus dem Stack heraus, damit bei späterer Verwendung des Blocks kein Fehler geschehen kann.

Da durch "CLST" der Stack auf Null gesetzt wird, kann ich diese mit dem nächsten Befehl auch gleich ins Register d abspeichern um alle Flags zu löschen.

Mit der Clr-Routine schalte ich auch gern gleich meinen Rechner aus, daher der Befehl "OFF".

Da es ja jetzt 2 Byte-Jumper gibt, werde ich beide Verfahrensweisen schildern, nach der man Sigma nach L bringen kann.

Zunächst die Arbeitsweise für den F1-Byte-Jumper.

```
Ø1 X()Y
Ø2 STO IND 22
Ø3 STO IND 25
Ø4 R up
```

Die vorstehenden Befehle werden in den Programmspeicher eingegeben. Wir gehen zur Zeile 2, verlassen den Pgrm-Modus und setzen den Byte-Jumper ein.

In den Pgrm-Modus zurückgeschaltet, sehen wir:

```
Ø2 ISG IND 17
```

Diese Zeile löschen wir mit der "Backarrow"-Taste. Jetzt sehen wir:

```
Ø1 X()Y
```

Wenn wir nun "DEL ØØ2" ausführen und anschließend einmal "SST", sehen wir die Zeile, die wir haben wollen:

```
Ø1 ΣREG L
```

Die Arbeitsweise mit dem F3-Byte-Jumper ist ähnlich, aber kürzer. Zunächst geben wir in den Programmspeicher

```
Ø1 1 E
Ø2 STO IND 25
Ø3 R up
```

ein. Beim Arbeiten mit dem F3-Byte-Jumper können wir ja im Pgrm-Modus bleiben. Darum gehen wir jetzt zurück zur Zeile Ø1, löschen sie mit der "Backarrow"-Taste und setzen den Byte-Jumper ein.

Wir tauschen also nur "1 E" gegen den Byte-Jumper aus.

Wir sehen jetzt:

```
Ø1T- [illegible]
```

Diese Zeile ist jetzt überflüssig und kann gelöscht werden. Nach SST sehen wir

    Ø1 Σ REG I

wie im 1. Fall.

Die Zeile

    STO d

kann in gleicher Weise geschaffen werden. Nur müssen wir hier im Fall 1

    Ø3 STO IND 17
    Ø4 AVIEW

und im 2. Fall die gleichen Befehle in den Zeilen 2 und 3 einsetzen.

Happy Programming

Niels Möhren (81)
Kielort 16
2000 Norderstedt
Telefon (040) 524 28 88
19.06.1981

Noch preiswerter !!

Clubmitglieder konnten bisher unter Angabe ihrer Mitgliedsnummer bei Firma Hansen in Bonn alles original HP-Zubehör und alle HP-Geräte zu 20 % unter Neupreis (unverbindl. empf. Listenpreis) erhalten. Seit kurzem besteht ein neues Angebot eines HP-Fachhändlers aus Hübingen. Er bietet 23% Rabatt bei voller Garantie und unter 500,—DM Bestellwert 5,--DM Portozuschlag.
Zahlung bei beiden Händlern per Vorkasse oder Nachnahme (NN).

Firma Annemarie Hansen	Dipl.Ing. Erwin Deutsch
z.Hd. Herrn Milles	Computertechnik Hübingen
Bachstr. 1-3	Hauptstr. 8
5300 Bonn 1	5401 Hübingen

HP-41 mit bis zu 16888 Bytes Programm/Datenspeicher

Auf gleichlautende Anzeige in CHIP 8-81 schrieben mir sogar zwei Clubmitglieder und erbaten ein Sonderinfo hierzu. Es handelt sich hierbei um den Einbau nahezu beliebig vieler QDMs in den HP-41c (beim cv nicht möglich) und deren Ansprechen über das sog. Page-Switching. Nähere Informationen hierzu von Hans Günther, der die Umbauten ausführt: siehe auch prisma 65-81

Primfaktorzerlegung

Ein weiters Programm ?!

Durch die Beiträge von Hauke Tetens (Prisma 208/81) und Jürgen Moeck (Prisma 209/81) wurde ich angeregt, auch mal ein Programm mit der in der Überschrift benannten Aufgabe zu schreiben, das zudem auch schnell sein sollte.

Die von Hauke genannte Zahl (6.869.821.302) wird von meinem Programm auf meinem Rechner in folgenden (gemessenen) Zeiten zerlegt.

Ohne angeschlossenen Drucker:

4 min. 18 sec.

Mit angeschlossenem, aber nicht eingeschaltetem Drucker:

4. min. 33 sec.

Mit angeschlossenem und eingeschaltetem Drucker:

7 min. 1 sec.

Ist jemand unter uns, der dieses Programm bei gleicher oder besserer Qualität noch schneller machen kann?

Bei der Erstellung des Programms habe ich Ideen nachstehend genannter Programmierer verwendet:

Friedrich Hillebrandt (HPACOR 10)
Richard Nelson (PPC 1)
John Kennedy (PPC 918)
Roger Hill (PPC 4940)
Phi Trinh (PPC 6171)

Nach der Eingabe des Programms in den Rechner wird es mit "n", XEQ "FF" gestartet. "n" steht hier für die Zahl, die zerlegt werden soll.

Ist eine Zahl zerlegt worden, oder als Primzahl ausgewiesen worden, kann das Programm mit "n", "R/S" neu gestartet werden.

Auf jede Zahl, die nicht zerlegt werden kann, sei es, daß sie kleiner als 2 ist; sei es, daß sie negativ ist, ein Bruch oder eine gemischte Zahl ist, reagiert der Rechner mit der Anzeige "DATA ERROR".

Auch in diesem Fall kann das Programm mit "n", "R/S" wieder neu gestartet werden.

Das Programm benötigt keine Datenregister. Es werden lediglich der Stack und das Alpha-Register verwendet.

Für die mögliche Benutzung des Druckers werden die Flags 21 und 25 manipuliert. Sonst werden keine Flags benutzt.

Gleich nach dem LBL "FF", noch vor der Eingangsroutine wird das "DATA ERROR" erzeugt, damit das Programm im entsprechenden Fall bequem neu gestartet werden kann.

In den Zeilen 11 bis 13 wird geprüft, ob ein Druch oder eine gemischte Zahl eingegeben wurde.

Statt der Befehlsfolge "ENTER", "INT" hätte ich auch "INT", "LASTX" nehmen können. Daß ich es nicht tat, hat einen bestimmten Grund.

Um den Befehl "LASTX" abzuarbeiten, Denötigt der Rechner etwa 35 msec., für den Befehl "ENTER" aber nur etwa 32 msec. Das ist eine ca. 9%ige Zeitersparnis.

Nun kann man ja sagen, 3 msec., was ist das schon. Das ist doch lächerlich. - Wir sollten aber immer darauf achten, mit Ausführungszeiten oder mit Bytes sparsam umzugehen, das kann uns bei großen und langen Programmen unter Umständen sehr zugute kommen.

Neulingen wird vielleicht die Zeile 31 auffallen. Ein alleinstehendes "E", was bedeutet das? Das alleinstehende "E" entspricht der Zahl "1". Das "E" ist ein Ergebnis sogenannter "synthetischer" Programmierung.

Was soll das aber? Hätte man die Zahl "1" nicht genausogut verwenden können? Doch, das hätte man. Hier spielt wieder die Zeitersparnis eine Rolle.

Wenn wir nach der Zeile 30 einmal auf die "EEX"-Taste drücken, kommt in Zeile 31 ein "1 E" zu stehen. Dieses "1 E" kann vom Rechner in einer um etwa 33,33 % kürzeren Zeitspanne abgearbeitet werden, als die Zahl "1".

Wir könnten natürlich das "1 E" stehen lassen, aber das kostet uns 2 Bytes, also doppelt so viel wie umbedingt nötig. Also nehmen wir die "1" vor dem "E" weg.

Diese Prozedur geht folgendermaßen vor sich. Zunächst gehen wir davon aus, daß die "synthetische" Funktion "F3-Byte-Jumper" einer Taste zugeordnet ist.

Jetzt geben wir folgendes ein. Zur besseren Orientierung nenne ich die Zeile 30 mit.

```
30 XEQ Ø3
31 STO 99
32 1 E
```

Wir gehen zurück zur Zeile 31, löschen den Befehl "STO 99" mit der "Backarrow"-Taste und ersetzen den gelöschten Befehl "STO 99" durch den Byte-Jumper.

Der Rechner muß dazu natürlich im USER-Modus sein.

Wir sehen jetzt die Zeile:

```
31 ‾▓▓
```

Der Byte-Jumper hat jetzt schon seine Pflicht getan und wir können diese Zeile auch wieder löschen. Einmal "SST" und wir sehen die gewünschte Zeile:

```
31  E
```

Was ist hier geschehen? Sowohl der Befehl "STO 99", als auch der Befehl "1 E" sind 2-Byte-Befehle. Wenn wir den ersten 2-Byte-Befehl "STO 99" löschen und den Byte-Jumper an seine Stelle setzen, schluckt er das 1. Byte des nachfolgenden 2-Byte-Befehls "1 E".

Der F3-Byte-Jumper ist auch ein 2-Byte-Befehl, und zwar der Zusammenschluß des Text-3-Befehls mit irgendeinem anderen 1-Byte-Befehl. Zum Beispiel, mit dem 1-Byte-Befehl "PROMPT".

Wenn wir eine Alpha-Zeile schreiben, ist die Byte-Anzahl immer Anzahl der Alpha-Zeichen plus 1.

```
"A" = 2 Bytes (F1, 41)
"AB" = 3 Bytes (F2, 41, 42)
"ABC" = 4 Bytes (F3, 41, 42, 43)
```

Der Text-Befehl gibt also die Anzahl der Alpha-Zeichen an, die in einer Zeile stehen.

Wenn wir einmal den Byte-Jumper allein eingeben, sehen wir die Zeile:

01 ‾E‾

Der Text-3-Befehl sagt dem Rechner: Es folgen 3 Alpha-Zeichen. Tatsächlich folgt aber nur ein einzelnes Byte.

Die Striche am oberen Display-Rand ist der Display-Ausdruck für das soganannte "Null"-Byte. Sozusagen der Ausdruck eines Vakuums. In das 2. Vakuum hinein wird nun also die "1" gerissen, das 1. Byte des 2-Byte-Befehls "1 E". Dadurch bleibt nun das "E" als alleiniger Zeilenausdruck stehen.

Wer den Byte-Jumper noch nicht besitzt, aber an ihm Interesse hat, den verweise ich an unseren Karten-Copy-Service (siehe Prisma 145-146/81) oder an einen unserer "Synthetic"-Experten, wie da sind zum Beispiel:

Matthias Grabiak
Rolf Mach
Andreas Marktscheffel

Alle siehe Mitgliederliste. Natürlich stehe auch ich gerne zur Verfügung.

Im Programmteil "LBL Ø3" wird der Stack gefüllt mit der zu zerlegenden Zahl, deren Quadratwurzel und einem Divisor.

In den Zeilen 42 bis 55 werden die notwendigen Divisionen durchgeführt und der Routinendurchlauf endet entweder, wenn der Divisor größer ist als die Quadratwurzel, oder wenn ein Divisor gefunden wurde, durch den ohne Rest geteilt werden konnte.

Die Zeilen 56 bis 61 beenden den Hauptteil des Programms mit einem Divisor "d" im X- und im L-Register und der Zahl "n" im Y-Register.

Happy Programming

Niels Möhren (81)
Kielort 16
2000 Norderstedt
Telefon (040) 524 28 88
14.07.1981

Programmlisting: Faktor Finder (von Niels Nöhren)

```
   PRP "FF"

01◆LBL "FF"
02 GTO 01

03◆LBL 00
04 CLX
05 /
06 LASTX

07◆LBL 01
08 CF 28
09 CF 29
10 FIX 0
11 ENTER↑
12 INT
13 X≠Y?
14 GTO 00
15 2
16 X>Y?
17 GTO 00
18 CLA
19 ARCL Y
20 "⊢="
21 XEQ 07
22 X=Y?
23 GTO 06
24 XEQ 03
25 X=Y?
26 GTO 06

27◆LBL 02
28 ARCL X
29 ST/ Y
30 XEQ 03
31  E
32 X=Y?
33 GTO 07
34 RDN
35 "⊢*"
36 GTO 02

37◆LBL 03
38 RCL Y
39 SQRT
40 LASTX
41 X<> Z

42◆LBL 04
43 X>Y?
44 R↑
45 R↑
46 X<>Y
47 MOD
48 X=0?
49 GTO 05
50 X<> L
51 2
52 X=Y?
53 SIGN
54 +
55 GTO 04

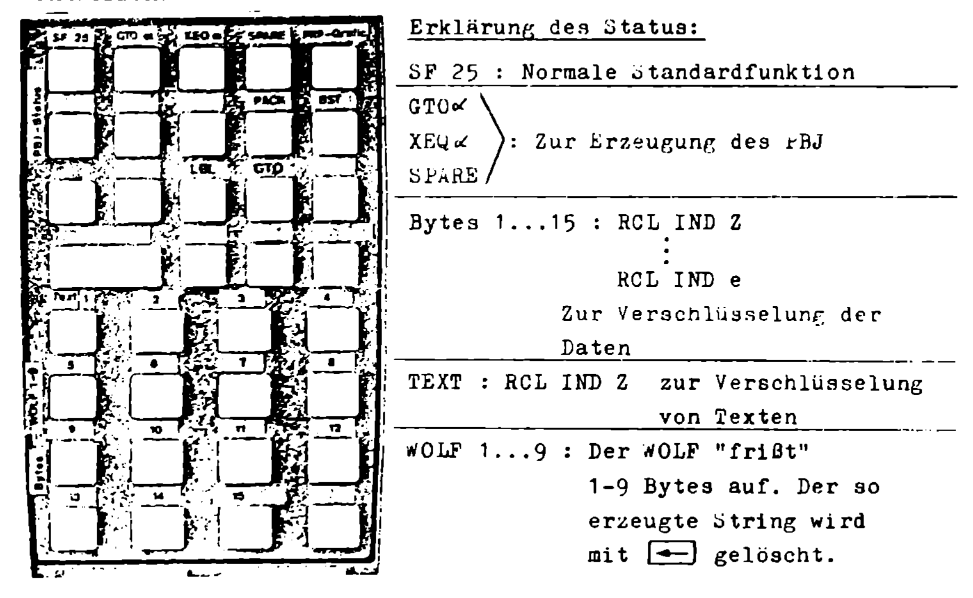
56◆LBL 05
57 R↑
58 LASTX
59 X>Y?
60 ENTER↑
61 RTN

62◆LBL 06
63 "PRIM"

64◆LBL 07
65 SF 25
66 PRA
67 SF 25
68 FS?C 21
69 CF 25
70 AVIEW
71 FC?C 25
72 SF 21
73 CLA
74 END
       CAT 1
LBL'FF
END
120 BYTES
```

P B J - Status

Ich habe eine PBJ-Statuskarte(programmierbarer Byte-Jumper) entworfen:



Erklärung des Status:

SF 25 : Normale Standardfunktion

GTO α, XEQ α, SPARE : Zur Erzeugung des PBJ

Bytes 1...15 : RCL IND Z
:
RCL IND e
Zur Verschlüsselung der Daten

TEXT : RCL IND Z zur Verschlüsselung von Texten

WOLF 1...9 : Der WOLF "frißt" 1-9 Bytes auf. Der so erzeugte String wird mit [←] gelöscht.

Verschlüsselung eines Programms:

[PRGM] [SF 25] [PBJ Ø1] [TEXT 1....15] [PRGM]

Der PBJ steht im Programm. Das SF 25 verhindert ein NONEXISTENT. Dann kommt der PBJ für ein Byte (PBJ Ø1) und ein Text-Byte, das die folgenden 1...15 Bytes zu Text macht. Dieses Textbyte wird in der Programmausführung vom PBJ übersprungen, so daß der Rechner trotz Textbyte die Bytes als Programmschritte ließt. Im Listing springt der Rechner nicht, so daß die Bytes als Text dargestellt werden. Das Programm ist somit verschlüsselt.

Eingabe des PBJ:

Beispiel:

:	Hex
54 Ø	1Ø
55 STO d	91 7E
56 ENTER↑	83
:	

das STO d in Zeile 55 soll verschlüsselt werden. Das Programm ist vorher in den Programmspeicher geladen worden.

```
54 0
55 STO d
56 ENTER↑
```

Eingabe:			Hex	
GTO .Ø54		Ø,ØØØØ		
PRGM on		54 Ø		
SF 25	(Σ+)	55 SF 25		Fehlerflag setzen
GTO∝	(1/X)	56 T-≰	F2 ØØ 29	3.Byte=1.Byte PBJ
LBL ØØ		57◆ LBL ØØ	Ø1	=2.Byte PBJ
Bytes 2	(7)	58 RCL IND Y	9Ø F2	2.Byte=Textbyte
BST		57 LBL ØØ		
BST		56 T-≰		
BST		55 SF 25		
WOLF 2	(■ 7)	56 T-2----■-		2 Byte (F2 ØØ) auffressen
←		55 SF 25		
SST		56 GTO T■		
PACK	(COS)	56 GTO T■		
WOLF 1	(■ -)	57 T-W----■		1 Byte (9Ø) auffressen
←		56 GTO T-		
PACK	(COS)	56 GTO T■		

---FERTIG---

PRGM:

⋮	Hex
54 Ø	1Ø
55 SF 25	A8 19
56 GTO T■	29 Ø1
57 T■Σ	F2 91 7E
58 ENTER↑	83
⋮	

```
54 0
55 SF 25
56 GTO ""
57 "Σ"
58 ENTER↑
```

Entschlüsselung der PBJ-geschützten Programme:

Die Programme, die bereits mit dem PBJ geschützt sind, lassen sich auf synthetischem Wege entschlüsseln.

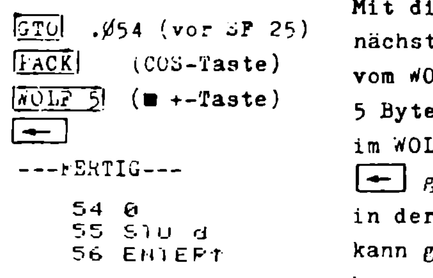
Beispiel:

⋮	Hex
54 Ø	1Ø
55 SF 25	A8 19
56 GTO T■	29 Ø1
57 T■Σ	F2 91 7E
58 ENTER↑	83
⋮	

Beim Entschlüsseln werden einfach die 5 Bytes A8 19 29 Ø1 F2 entfernt.

```
54 0
55 SF 25
56 GTO ""
57 "Σ"
58 ENTER↑
```

Eingabe:

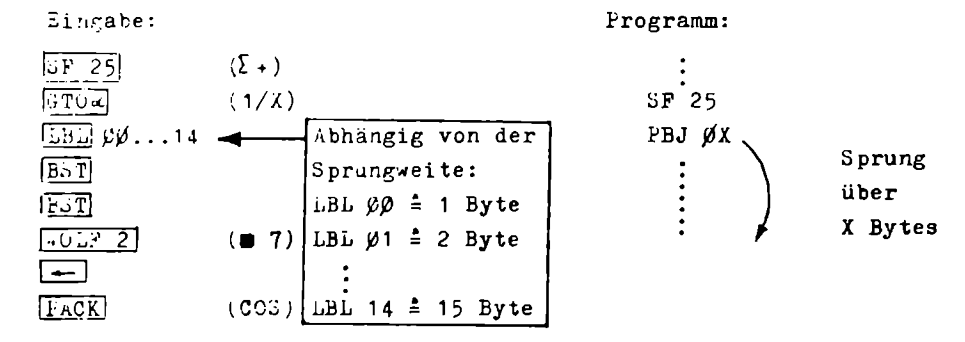


Mit dieser Befehlsfolge werden die nächsten 5 Byte beginnend mit Zeile 5 vom WOLF 5 "aufgefressen". Diese 5 Bytes stehen dann als ALPHA-Zeichen im WOLF-String. Dieser wird dann mit [←] gelöscht. Das Programm steht dann in der Ursprungsversion zur Verfügung, kann geändert werden, und bei Bedarf kann man es später wieder verschlüsseln.

Der PBJ mit XEQ (und SPARE ???) funktioniert ähnlich.

Sprünge mit dem PBJ:

Mit dem PBJ kann man relative Sprünge um 1...15 Bytes programmieren. Die Methode ist dieselbe wie bei der Verschlüsselung mit dem Unterschied, daß das 2. Byte im Byte-Jumper (bei der Verschlüsselung Ø1, erzeugt durch LBL ØØ) verändert wird (LBL ØØ...LBL 14).



Erzeugung des PBJ-Status über das Key-Assignment-Programm:

Achtung!!! Genügend freie Register schaffen. Wenn KA im PRGM-Speicher ist, müssen nach GTO .. noch mindestens 15 freie Register sein.

Eingaben ...'s K-Programm: 15 PAIRS

KEY	1. Byte	2. Byte	Taste
1	168	25	11
2	242	29	12
3	242	3Ø	13
4	242	31	14
5	4	1Ø	24
6	4	7	25
7	144	241	51
8	144	242	52
9	144	243	53
1Ø	144	244	54
11	144	245	61
12	144	246	62
13	144	247	63
14	144	248	64
15	144	249	71
16	144	25Ø	72
17	144	251	73
18	144	252	74
19	144	253	81
2Ø	144	254	82
21	144	255	83
22	247	87	-51
23	248	5Ø	-52
24	249	51	-53
25	25Ø	52	-54
26	251	53	-61
27	252	54	-62
28	253	55	-63
29	254	56	-64
3Ø	255	57	-71

Happy Programmjumping

Hagen

Ich habe ein Programm geschrieben, mit dem man Großschrift schreiben kann. Das Programm heißt POWER-LETTER und ist beim Magnetkartenservice zu erhalten. Es füllt 18 Kartenseiten. Es ist der komplette Zeichensatz und zusätzlich Grafic möglich. Tastenzuordnungen sind möglich (synthetisch). Wer Interesse an Statuskarten und Programm hat, schicke bitte 11 Magnetkarten an den Magnetkartenservice. Bedingt durch die Länge des Programms ist ein HP-41CV oder ein HP-41C mit 4 Memory-Modulen notwendig.

Hagen Klemp (73)

HEWLETT-PACKARD ANWENDER-CLUB
OLIVER RIETSCHEL

ABTEILUNG : KARTENCOPYSERVICE

Ralf Pfeifer
Rubensstr. 5
5000 Koeln 50
Tel. 0221/352034

Walter Pieperhoff
Walb. Osth. Wallstr. 10
4770 Soest
Tel. 02921/2895

vorhandene Programme Liste Nr. 2	Anzahl d. Karten
46. ABL von Christof Born	1
47. LIN " "	1
48. BIN " "	1
49. DEZ " "	1
50. CL " "	1
51. FN " "	1
52. FFZ optimiert " von N. Noehren	1
53. Kalender Ausdruck von "	3
54. Bastel Prg. (Wallkarten) Rechner Nr.< 1950	5
55. Bastel Prg. fuer alle Rechner	1
56. BLD von Matthias Grabiak	1
57. BLD (Drucker) "	1
58. CNT "	1
59. VSZE "	1
60. *I "	1
61. FFZ "	1
62. Amateurfunk - Kenner	3
63. Formatierter Ausdruck	1
64. Standartabweichung	1
65. Matizenrechnung Programmpaket	7
66. B3 = Loeschen von Flag 55	1
67. a↑2 + b↑2 = Primzahl	1
68. Filmlaengen - Bestimmung	2
69. Plottprogramm	2
70. Batterietest	1
71. ACXR optimiert v. W. Kropf	1
72. Dreiecksberechnung	1
73. Monatslohnsteuer	6
74. Interpolation	4
75. Car Control	4
76. Stern - Dreieck Umrechnung	1
77. Addiert Sinusschwingungen	1
78. Flags Anzeige	1
79. Ringmatrix	1
80. Magisches Quadrat	1
81. Hex - Dezm. Wandler	1
82. Raten	1
83. Multiplotter verbessert	3
84. XROM von Ralf Pfeifer	1
85. STAT "	1
86. GET "	1
87. GETA "	1
88. GAMMA "	1
89. Lohnverteilung	2
90. Black - Jack	2

Bei Bestellungen : erforderliche Kartenanzahl u. frankierten adressierten Briefumschlag zuschicken.

HEWL[illegible]-PACKARD ANWENDER-CLUB
OLIVER RIETSCHEL

ABTEILUNG : KARTENCOPYSERVICE

Ralf Pfeifer	Walter Pieperhoff
Rubensstr. 5	Walb. Osth. Wallstr. 10
5000 Koeln 50	4770 Soest
Tel. 0221/352034	Tel. 02921/2895

vorhandene Programme Liste Nr. 3	Anzahl d. Karten
91. Mantisse m der Zahl in X	1
92. Dez. - Hex. Wandler v. Juergen Holm	1
93. Hex. - Dez. Wandler	1
94. Size Routine (Schnellstmoeglich)	1
95. Ohm'sche Daempfungsglieder	1
96. Auswertung aussagenlogischer Ausdruecke	1
97. Key Assingment v. Andreas Marktscheffel	1
98. Algem. Prg. zu CHIP Preisausschreiben	1
99. Flugnavigation	4
100. Primfaktorzerlegung R. Pfeifer/U. Davertsh	1
101. Zeitschleife	1
102. Hornerschema	1
103. Lohnsteuer (Oesterreich)	3
104. Euklid	1
105. Fermatsche Methode	1
106. Primfaktorzerlegung Friedrich Hillebrandt	4
107. Sortieren von 3 Werten	1
108. Kegel Programm	2

Ich(Walter Pieperhoff) habe beobachtet ; dass besonders Synthetische Programme beim Copyservice bevorzugt werden und biete deshalb aus dem Buch 'Synthetic Programming von W. C. Wickes folgendes an :

Programm	Karten
109.LBL'AD'zeigt nach RCL b die Adresse in Hex an	1
110.LBL'AL'tauscht 6 Alphaz. v. Reg. in Y nach Reg. in X	1
111.LBL'BY'zeigt nach RCL b die Adresse Dezimal an	1
112.LBL'CA'loescht alle Tastenzuordnungen	1
113.LBL'CD'wandelt 1 Alphazeichen in Hexcode um	1
114.LBL'CODE'wandelt 14 Hexcodes in Alphazeichen um	1
115.LBL'CU'wandelt von Reg. 00 an in Programmregister um	1
116.LBL'DC'wandelt 1Hexcode in Alphazeichen um	1
117.LBL'DECODE'wandelt 7 Alphazeichen in Hexadezimal um	1
118.LBL'DI'zeigt alle Segmente im Display(Displaykontr.)	1
119.LBL'EF'ermittelt noch freien Programmspeicher	1
120.LBL'EN'ermittelt das Ende des letzten Programms im Sp	1
121.LBL'FL'setzt Flag Nr. in X	1
122.LBL'HM' Wortraten synthetisch	2
123.LBL'ISO'isoliert bez.Buchst. in X aus 24 in Alphareg.	1
124.LBL'KA' Key - Assingment	1
125.LBL'KP'packt die Key - Assingnment Register	1
126.LBL'S' zeigt SIZE an	1
127.LBL'TONE'erzeugt 128 Synthetische Toene	1
128.LBL'SAVE'speichert Flags in Reg. 00 + Reg.01 ab	1
129.LBL'RE' holt Flags aus Reg. 00 + Reg. 01	1
130.LBL'MANT' zeigt die Mantisse an	1
131.LBL'REV'dreht 6Alphazeichen im Alphareg. um	1
132.LBL'REG'speichert mit CODE erzeugte Z. in Reg.Nr.in X	1

Bei Bestellungen : erforderliche Kartenanzahl u. frankierten adressierten Briefumschlag zuschicken. ...

Anmerkungen zum Copy-Service

1. Immer wieder landen bei mir Sendungen, die einen unterfrankierten Rückumschlag beinhalten. Wer also Karten bestellt, sollte diese in seinen Rückumschlag legen und wiegen. Da Walter und ich noch ein Blatt mit Hinweisen und Bestätigung der Bestellung beilegen, bitten wir darum, auch dieses Gewicht beim wiegen zu berücksichtigen. Ich überprüfe erst jeden Umschlag auf sein richtiges Gewicht, und nur falls es stimmt kommt mein Absenderstemel dar auf. Somit muß jeder Besteller sein Strafporto selbst bezahlen !!!
2. Nach einigen unsicheren Anfragen zu schließen, scheint nicht jedem klar geworden zu sein, daß der Besteller auch eine entsprechende Menge an Magnetkarten beilegen muß ! Sie werden n i c h t von irgend jemandem gestiftet !
3. Vielen Dank allen denen, die Briefmarken oder Magnetkarten oder gar Geldbeträge stiften wollen oder gestiftet haben (Geld kam immer zurück). Aber es ist natürlich nur ein kostenloser Service !
4. Um evl. Gerüchten vorzubeugen: Wir erhalten vom Club leihweise Magnetkarten, auf die wir die Programme abspeichern ! Alles andere erledigen wir ehrenamtlich und gratis (Kartenleserabnutzung, Zeit, Strom).
5. Einen Kartencopy-Service, der den Abonennten zu jeder prisma die Karten liefert, wird es wohl nicht geben. Grund: Nur 1 Interessent !
6. Bitte, beachtet die Schulferien in Nordrhein-Westfalen ! Während dieser Zeit bin ich nicht gerade oft auf deutschem Staatsgebiet anzutreffen! Eure Kartenbestellungen liegen dann nur bei mir zu Hause herum !

Ralf Pfeifer

Hinweis zu Search and Destroy

Dieses Programm wird wohl nicht in der Liste des Copy-Srevice auftauchen, da es ja nur für WAND-Besitzer zu gebrauchen ist, und die es schnell selbst laden können.
Wer sich dagegen wundert, daß er mit diesem Programm auf keinen grünen Zweig kommt, der sollte sich mal den Barcode über der Zahl 70 auf dem Spielfeld ansehen: Er ist fälschlicherweise der gleiche wie bei 60 !!

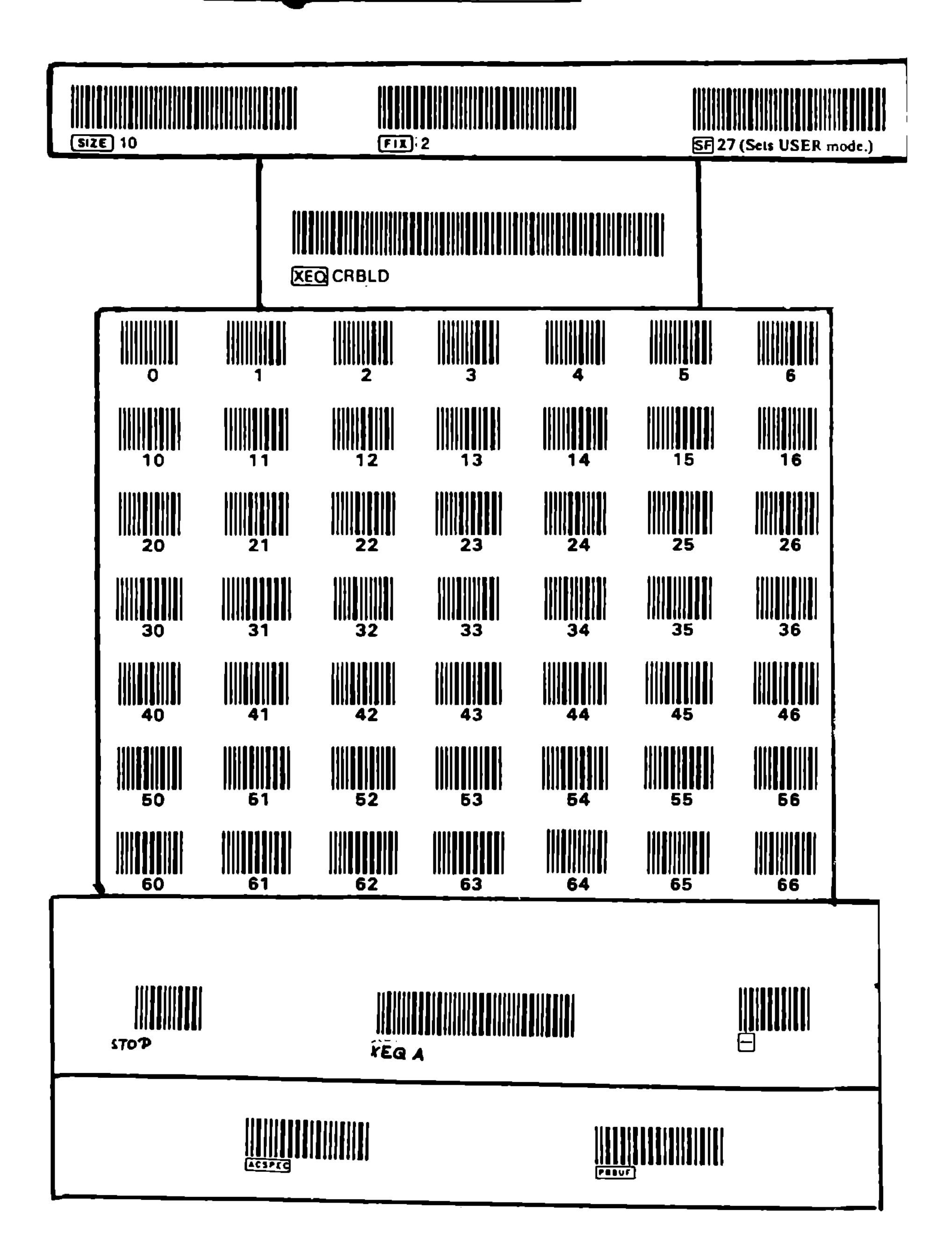
Ralf Pfeifer

Verbesserung des "Character Builder"

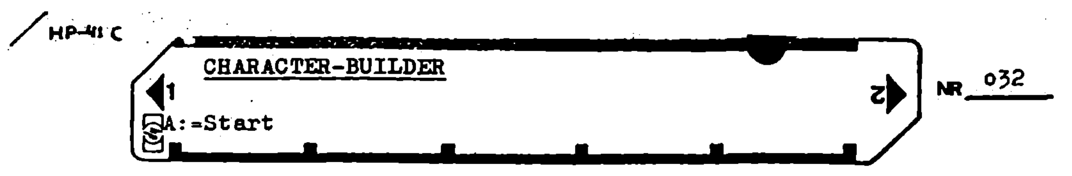
Wie mancher WAND-Besitzer sicher mit Verärgerung festgestellt hat, ist dieses Programm für die Druckeranwendung nahezu wertlos; Denn der Drucker hat eine 7x7-Matrix, das Programm aber nur eine 6x6. Und das, weil das Programm eine vollkommen überflüssige -gleichwohl ästhetische- Großanfertigung des Zeichens ausdruckt.
Eine Besserung dieses Mißstandes verspricht das beigefügte Programm: Auf dem ebenfalls beiliegenden Blatt kann man ganz einfach sein Sonderzeichen mit dem WAND nachzeichnen und ausdrucken lassen.
Dem Komfort dienlich ist auch die Eigenschaft des Programms, zu überprüfen, ob ein Punkt schon einmal eingegeben wurde. Falls ja, löscht das Programm den Punkt und piepst dazu. Dieses Spiel von Löschen und Setzten (also Punkt drucken oder nicht drucken—wie bei einem Flag) kann man beliebig oft wiederholen.

Ralf Pfeifer

CHARACTER BUILDER -- KEY-PAD



HP-41C



NR 032

Program Title / Programmtitel / Titre du programme / Titolo del programma	Character Builder
Name / Name/Nom/Nome	Ralf Pfeifer
Address / Strasse/Adresse/Indirizzo	Rubensstr. 5 5000 Köln - 50 Tel. 35 20 34
City / Ort / Localité / Città	
Postal Code / Postleitzahl / Code postal / C.A.P.	
Country / Land / Pays / Paese	BR Deutschland
No. of program lines / Anzahl Programmzeilen / Nombre de lignes de programme / N. di linee di programma	071 LINES 112 BYTES
No. of data registers / Anzahl der benutzten Datenspeicher / Nombre de registres de données / N. di registri utilizzati	010
HP Applications ROM / HP Applikations ROM / ROM d'application HP / ROM di applicazione HP	WAND 82153A Thermodrucker 82143A
Program Name: / Programm: / Nom du programme: / Programma:	

Program Abstract / Kurzbeschreibung / Resume / Breve descrizione del programma: Das Programm ist eine Verbesserung des gleichnamigen Handbuchprogramms. Bei diesem Programm steht eine 7x7-Matrix zur Erstellung der Sonderzeichen zur Verfügung. Zur Anwendung benutzt man zweckmäßigerweise das beiliegende "Character Builder-Key-Pad".

Operating limits and Warnings / Grenzen und Einschränkungen / Limites et restrictions / Limiti operativi e avvertenze: Hat der gelesene Barcode mehr als 1 Byte, startet sich das Programm erneut. Barcodes, deren dezimaler Wert eine 7,8 oder 9 enthält oder größer als 66 ist, führen dazu, daß die Eingaben zu einem Sonderzeichen zusammengestellt werden. Während die WNDSCN-Funktion arbeitet, können Befehle nur nach einem R/S über die Tastatur eingegeben werden.

FLAGS	SET STATUS		
	FLAGS (ON OFF)	TRIG	DISP
	0 ☐ ☐	DEG ☐	FIX ☐
	1 ☐ ☐	GRAD ☐	SCI ☐
	2 ☐ ☐	RAD ☐	ENG ☐
	3 ☐ ☐		n___

MAN TRACE NORM ☒ On ☐ Off USER

1	2
3 WAND	4 PRINTER

INSTRUCTIONS D'EMPLOI I
NORME OPERATIVE I

Ralf Pfeil

Step	Instructions	Variables	Function(s)	Result
1	WAND und PRINTER anschließen.			
2	Datenspeicher wählen		SIZE 010	
3	USER-Modus wählen.		SF 27	
4	Drucker ausschalten (Geschw.Vort.)			
5	KEY-PAD zurechtlegen.			
6	Programm starten:		XEQ"CRBLD"	
		oder:	XEQ A	
		oder:	RTN R/S	
7	Nun fordert der Rechner auf,			W: READY
	einen Barcode zu lesen.			
8	Auf dem KEY-PAD lesen Sie nun den			
	Barcode, der dem von Ihnen ge-			
	wünschten Punkt entspricht.			
9	Sobald der dezimale Wert des			
	letzten gelesenen Barcodes in der			
	Anzeige erscheint, kann ein neuer			
	gelesen werden.			
10	Zurück zu 8.			
11	Um die Eingabe zu beenden, und			
	das Sonderzeichen erstellen zu			
	lassen, STOP oder - lesen.			
	Das Zeichen kann nach Programm-			
	ende mit ACSPEC und PRBUF inden			
	Druckbuffer geladen und gedruckt			
	werden.			
12	Korrektur: Falls unter 7 oder 9			
	ein Barcode irrtümlich gelesen			
	wurde, genügt es, wenn Sie ihn			
	ein 2. mal mit dem WAND eingeben.			
	Der Rechner piepst dann zur Bestätigung			
	der Korrektur.			
	Dieses Spiel von einlesen und			
	löschen läßt sich -auch mit dem			
	gleichen Barcode- beliebig oft			
	wiederholen.			
	Falls das Programm irrtümlich			
	mit der errechnung des Sonder-			
	zeichens beginnt, weil entweder			
	ein Barcode der Matrix falsch er-			
	kannt oder versehentlich/absicht-			
	lich STOP gelesen wurde, kann			
	die Eingabe durch		XEQ 00	
	wieder gestartet werden.			
	Der Anwender darf jedoch die In-			
	halte der Register 02-08 nicht			
	verändern!			

BEISPIEL:

EINSCHALTEN EINES GLEICHSTROM-MOTORS MIT ANKERRUECKWIRKUNG

$$\frac{dw}{dt} = \frac{c*i}{I}$$

$$\frac{di}{dt} = \frac{U-R*i-c*w}{L}$$

```
 01*LBL "dO"        ← F()
1,2  RCL 22  40  /  X↑2
1  +  /  STO 25  RCL 22
*  .05  /  RTN

 16*LBL "dI"        ← G()
225  RCL 22  3,8  *  -
RCL 25  RCL 14  *  -
,10  /  END
```

```
      PLOT OF DIFFGL
 t,X (UNITS= 1) ↓
 ω,Y (UNITS= 1) →
      0              225
      0
      ├──────────────────┤
0,00 • :
0,05  ·              :
0,10     ·             :  ← I
0,15       ·           :
0,20          ·        :
0,25            ·      :
0,30              ·   :
0,35                ·:
0,40               :  ·   ← ω
0,45            :       ·
0,50        :            ·
0,55    :                ·
0,60   :                 ·
0,65   :                ·
0,70   :               ·
0,75   :               ·
0,80   :               ·
0,85   :               ·
0,90   :               ·
0,95   :               ·
1,00   :               ·
 I,Y (UNITS= 1) →
   -11,0            53,6
      0,0
   ├─┼─────────────────┤
```

PRINTEN DURCH DIFFGL:

```
X0= 0,000
Y0= 0,000
Z0= 0,000
X1= 0,050
Y1= 16,944
Z1= 37,123
X2= 0,100
Y2= 40,698
Z2= 48,619
X3= 0,150
Y3= 64,065
Z3= 51,179
X4= 0,200
Y4= 87,330
Z4= 50,280
X5= 0,250
Y5= 110,830
Z5= 47,730
```

U S W .

PRINTEN EXTERN:

```
 01*LBL "AUX"
XEQ "DIFFGL"  "t"  8
XEQ 16  "├ sec"  AVIEW
"w"  12  XEQ 16
"├ rad/s"  AVIEW  "I"
20  XEQ 16  "├ A"  AVIEW
GTO "AUX"

 19*LBL 16
FIX 0  ARCL 19  "├= "
RCL 10  LOG  1  +
FIX IND X  ARCL IND Y
END
```

```
t0= 0,000 sec
w0= 0,000 rad/s
I0= 0,000 A
t1= 0,050 sec
w1= 16,944 rad/s
I1= 37,123 A
t2= 0,100 sec
w2= 40,698 rad/s
I2= 48,619 A
t3= 0,150 sec
w3= 64,665 rad/s
I3= 51,179 A
t4= 0,200 sec
w4= 87,330 rad/s
I4= 50,280 A
t5= 0,250 sec
w5= 110,800 rad/s
I5= 47,730 A
```

DIESES BEISPIEL BESITZT ORDNUNG 2. FOR BEISPIELE MIT ORDNUNG 1 ODER 4 EMPFEHLE ICH SIN UND COS FUNKTIONEN.
ANSONSTEN VERWEISE ICH AUF DIE LITERATUR.

HAPPY PROGRAMMING

PLOTTEN VON 2 FUNKTIONEN
... PUNKTEN

PLOT2 STELLT DIE DATEN ZUM PLOTTEN VON 2 PUNKTEN BEREIT. AM ENDE DES PLOTS WIRD DIE ACHSE VON Y2 BESCHRIFTET.
PRINTERSCHALTER: MAN

PROGRAMMSEQUENZ:

```
  LBL ... (UNTERPROGRAMM
  ....     (FUER "PRPLOT"
  RCL Y1
  RCL Y2
  XEQ PLOT2
  END          ODER REGPLOT
                  ....

                  ....
```

VORBEDINGUNG

PLOT2 MUSS IM USER-MODE GESTARTET WERDEN. WIRD GELOESCHT VOR EINGABE.

STATUS:

GLEICHE REG WIE PRPLOT:
R02:nnn.aY2,R03:YMIN2
R04:YMAX2,F27:EINGABE?

```
     PLOT OF DEMO
   X (UNITS= 1) ↓
   Y (UNITS= 1) →
    0,0            21,0
    0,0
   |-----------------|
 0                  ·:
 24              ·   :
 48             ·  :
 72            · :
 96          :·
120        :  ·
144  :       ·
168:       ·
192:      ·
216  :    ·
240     ·:
264    ·     :
288   ·
312  ·           :
336  ·             :
360 ·               :
   Y (UNITS= 1) →
    -1,00        1,00
          0,00
   |--------|--------|
```

```
01♦LBL "PLO
        T2"
02 FC?C 27
03 GTO 00
04 "YMIN2?"
05 PROMPT
06 STO 03
07 RDN
08 "YMAX2?"
09 PROMPT
10 STO 04
11 RDN
```

EINGABE, MUSS DAS ERSTE MAL DURCHLAUFEN WERDEN.

```
12♦LBL 00
13 RCL 03
14 -
15 RCL 04
16 RCL 03
17 -
18 /
19 RCL 02
20 INT
21 STO 02
22 *
23 FIX 0
24 RND
25 1 E3
26 /
27 ST+ 02
28 X<>Y
29 RCL 09
30 RCL 10
31 RCL 06
32 +
33 X>Y?
34 GTO 01
35 R↑
36 RTN
```

BEREITSTELLEN DER DATEN FUER REGPLOT.

```
37♦LBL 01
38 R↑
39 REGPLOT
40 RCL 03
41 STO 00
42 X<=0?
43 CLX
44 X<> 04
45 STO 01
46 X<0?
47 STO 04
48 XROM "PR
      AXIS"
49 BEEP
50 "FIN"
51 PROMPT
52 END
```

LETZTE SPALTE UND ZWEITE Y-ACHSE UND SCHLUSSHALT.

ERSTES PLOT-PROGRAMM FÜR 2 FUNKTIONEN MIT VERSCH. Y-ACHSEN!

PLOT2 VERWENDET DIE X-ACHSE ZUM PLOTTEN DER 2. FUNKTION. IN DER ERSTEN Y-ACHSE KANN EIN MARKIERUNGSSTRICH FREI GEWÄHLT WERDEN (z.B. NULLAGE). IN DER ZWEITEN Y-ACHSE WIRD DIE NULLAGE ODER DER WERT NÄHER BEI NULL AUTOMATISCH MARKIERT.
PLOT2 BENÖTIGT KEINE ZUSÄTZLICHEN SPEICHERREG. IN R02 IST DIE LAGE DES PLOTPUNKTES UND DER X-ACHSE FESTGELEGT.
R03 BESTIMMT DEN PLOT-CHARACTER WENN NICHT NUM UND R04 ZEIGT, DASS DIE ACHSE GEPLOTTET WERDEN SOLL, WENN NUMERISCH.
DAS PLOT2-PROGRAMM ENTSTAND ALS NEBENPRODUKT VON DIFFGL, DA DORT ZWEI FUNKTIONEN MIT VERSCH. Y-ACHSEN GEPLOTTET WERDEN SOLLTEN.

Happy Programming
Max Huber (1986)

```
LBL'PLOT2
END
87 BYTES
```

```
 01♦LBL "DEMO"
COS  LASTX  SQRT  CHS
20  +  X<>Y  XEQ "PLOT2"
END
```

Berlin [illegible]en 22.5.81

BALKENDIAGRAMM-PLOTTER

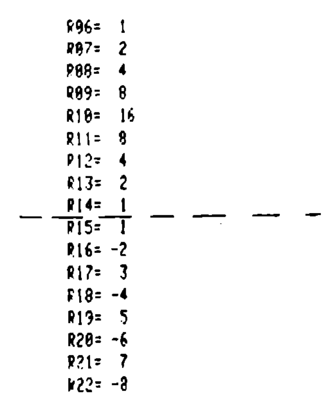
Dieses Programm ist in der Lage, Balkendiagramme der Inhalte einer Reihe von vom Anwender angegebenen Datenrgister auszudrucken. Dabei besteht auch die Möglichkeit, positive und negative Werte im selben Diagramm darzustellen.

Anwendung:

1. XEQ"COL"	Anzeige: "Y MIN?"
2. Minimalen Wert für y eingeben; R/S	" : "Y MAX?"
3. Maximaler Wert für y; R/S	" : "REG MIN?"
4. Kleinste Registeradresse für Ausdruck eingeben(>5)	" : "REG MAX?"
5. Endadresse eingeben	" : "LABELS +/-?
6. Hier muß eine Zahl eingegeben werden, die später zu jeder Registeradresse addiert wird, um die einzelnen Balken zu kennzeichnen. R/S	" : "TEXT?"
7. Hier kann ein Text eingegeben werden, der dann vor dem Diagramm ausgedruckt wird. Wird nichts eingegeben, so wird eine Leerzeile ausgedruckt.	(Ausdruck fängt an)

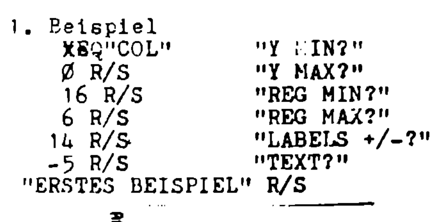
2 Beispiele:

Die folgenden Daten eingeben:

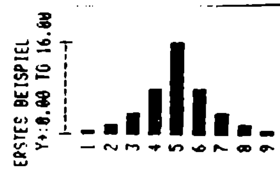


R06= 1
R07= 2
R08= 4
R09= 8
R10= 16
R11= 8
R12= 4
R13= 2
R14= 1
R15= 1
R16= -2
R17= 3
R18= -4
R19= 5
R20= -6
R21= 7
R22= -8

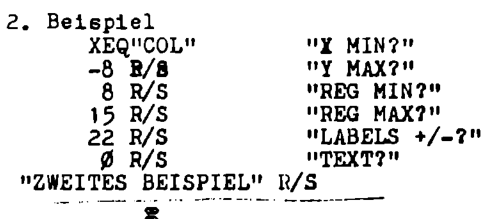
1. Beispiel



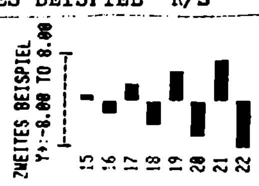
XEQ"COL"	"Y MIN?"
Ø R/S	"Y MAX?"
16 R/S	"REG MIN?"
6 R/S	"REG MAX?"
14 R/S	"LABELS +/-?"
-5 R/S	"TEXT?"
"ERSTES BEISPIEL" R/S	



2. Beispiel



XEQ"COL"	"Y MIN?"
-8 R/S	"Y MAX?"
8 R/S	"REG MIN?"
15 R/S	"REG MAX?"
22 R/S	"LABELS +/-?"
Ø R/S	"TEXT?"
"ZWEITES BEISPIEL" R/S	



```
01◆LBL "COL
"
02 CF 00
03◆LBL 02
04 "Y MIN?"
05 PROMPT
06 X>0?
07 GTO 02
08 STO 01
09 "Y MAX?"
10 PROMPT
11 STO 02
12 X<=Y?
13 GTO 02
14 "REG MIN
?"
15 PROMPT
16 STO 04
17 "REG MAX
?"
18 PROMPT
19 STO 05
20◆LBL 03
21 "LABELS
+/-?"
22 PROMPT
23 STO 03
24 RCL 05
25 +
26 1 E3
27 X<=Y?
28 GTO 03
29 "TEXT?"
30 CF 21
31 AVIEW
32 AON
33 CLA
34 STOP
35 AOFF
36 SF 21
37 PRA
38 FIX 2
39 "  Y→:"
40 ARCL 01
41 "├ TO "
42 ARCL 02
43 ACA
44 PRBUF
45 4
46 SKPCHR
47 127
48 ACCOL
49 1
50 SKPCOL
51 10
52 "-"
53◆LBL 04
54 ACA
55 DSE X
56 GTO 04
57 1
58 SKPCOL
59 R↑
60 ACCOL
61 PRBUF
62 RCL 02
63 RCL 01
64 -
65 LASTX
66 ABS
67 X<>Y
68 /
69 74
70 *
71 STO 00
72 FIX 0
73 CF 29
74◆LBL 05
75 RCL 04
76 RCL 03
77 +
78 CLA
79 10
80 X>Y?
81 " "
82 ST* X
83 X>Y?
84 "├ "
85 ARCL Y
86 ACA
87 1
88 SKPCHR
89 RCL IND
04
90 X<0?
91 GTO 06
92 RCL 02
93 /
94 1
95 X<=Y?
96 SF 00
97 X>Y?
98 X<>Y
99 74
100 RCL 00
101 X≠0?
102 SKPCOL
103 -
104 *
105 XEQ 00
106◆LBL 07
107 FC?C 00
108 PRBUF
109 RCL 04
110 1
111 ST+ 04
112 RDN
113 RCL 05
114 X>Y?
115 GTO 05
116 RTN
117◆LBL 06
118 RCL 01
119 ABS
120 X<>Y
121 ABS
122 -
123 RCL 02
124 RCL 01
125 -
126 /
127 74
128 *
129 X<0?
130 0
131 SKPCOL
132 RCL 00
133 X<>Y
134 -
135 XEQ 00
136 GTO 07
137◆LBL 00
138 INT
139 RCL X
140 2
141 /
142 INT
143 SF 12
144 XEQ 01
145 RDN
146 2
147 MOD
148 CF 12
149 XEQ 01
150 RTN
151◆LBL 01
152 X=0?
153 RTN
154 127
155 ACCOL
156 RDN
157 DSE X
158 GTO 01
159 END
```

```
               GTO ..
PACKING
               CAT 1
LBL'COL
END        287 BYTES
```

Bar-Codes & Graphik mit dem HP-41C

Welcher Druckerbesitzer hat sich nicht schon darüber geärgert,daß er keinen Einfluß auf den Vorschub seines Druckers hat.Die schönsten Graphiken und "High Resolution Plottings" werden durch den Zeilenzwischenraum verschandelt.Unter größten Anstrengungen bekommt man gerade 2 Bytes als Barcode waagerecht auf den Papierstreifen und bei Versuchen mit senkrechtem Druck sagt der Griffel entweder garnichts,oder "CHECKSUM ERR" oder eine Programmcodezeile erreicht die beachtliche Länge von ca. 50cm!Sollte man sich vielleicht doch einen "richtigen" Computer kaufen?

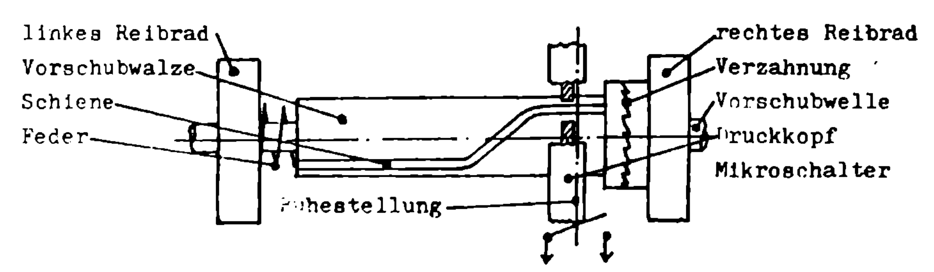
Die Lösung des Problems liegt in einem mechanischen Umbau des Druckers,der diesen dazu bringt,nur noch den halben Zeilenvorschub auszuführen.Die Druckzeilen überlappen sich dann um eine Linie.Der Umbau wird so ausgeführt,daß der Drucker zwischen dem normalen und dem halben Vorschub umschaltbar ist,die normalen Eigenschaften bleiben also erhalten.Zu beachten ist,daß bei halbem Vorschub nur noch 133 statt 168 Druckspalten in eine Zeile gehen. Dies beruht darauf,daß die gesamte Druckzeile um ca. 12mm auf dem Papierstreifen nach links verscoben wird.

Benötigt werden für den Umbau:
- ein zweiter Mikroschalter vom gleichen Typ,wie er schon im Drucker vorhanden ist,
- ein einfacher 1xEin Schalter zum Umschalten des Vorschubes
- Draht,Lötkolben,Lötzinn,Schlüsselfeile,usw.,
- ein wenig Fingerspitzengefühl,technisches Verständnis und mindestens 2 Stunden Zeit.

Belohnt wird man dafür mit fast perfekter Graphik und gut lesbaren Bar-Codes,die dem Standardformat von HP sehr nahe kommen.Die Garantie ist allerdings verlohren;für Schäden,die durch den Umbau entstehen kann ich keinerlei Haftung übernehmen.

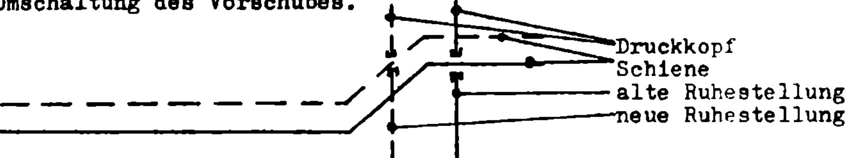
Der normale Druckervorschub

Der Drucker HP 82143A hat einen mechanischen Vorschub, der eine stufenlose Regelung unmöglich macht.Gesteuert wird er von einer auf der Vorschubwalze befindlichen Schiene und einem Mikroschalter,der die Ruhestellung festlegt.Während sich der Druckkopf nach links bewegt,wird die Zeile ausgedruckt,die Vorschubwalze dreht sich nach oben und überspringt 2 Zähne in der Verzahnung.Die beiden durch die Vorschubwelle miteinander verbundenen Reibräder stehen währenddessen still und sorgen dafür,daß sich der Papierstreifen nicht bewegt.Nach Ausdruck der Zeile bewegt sich der Druckkopf wieder nach rechts und dreht die Vorschubwalze über die Schiene wieder nach unten bis er den Kontakt im Mikroschalter schließt und der CPU damit mitteilt,daß er sich in Ruhestellung befindet.Der Motor wird abgeschaltet,der Druckvorgang für diese Zeile ist beendet.Die Drehung der Vorschubwalze wurde über die Verzahnung auf die Vorschubwelle und damit auf die Reibräder übertragen,die wiederum den Papierstreifen um ca. 4mm nach oben bewegten.



Der veränderte Druckervorschub:

Verschiebt man den Mikroschalter nach links,so verschiebt sich auch die Ruhestellung und damit die Ganze Druckzeile nach links.Wenn man die Ruhestellung weit genug nach links verschiebt kann nur noch ein Zahn in der Verzahnung übersprungen werden,der Vorschub wird halbiert.Nun schlägt jedoch der Druckkopf nach 133 Druckspalten am Gehäuse an und eine eingebaute Überlastungssicherung schaltet den Motor aus.Nach Aus- und Einschalten des Druckers wird die Sperre aufgehoben.Um nun alle Möglichkeiten des Druckers zu erhalten bauen wir zwei Mikroschalter ein,einen in der alten Ruhestellung für normalen Vorschub und einen in der neuen Ruhestellung für halben Vorschub.Ein weiterer,von außen zugänglicher Schalter erlaubt die Umschaltung des Vorschubes.



Umbaubeschreibung:

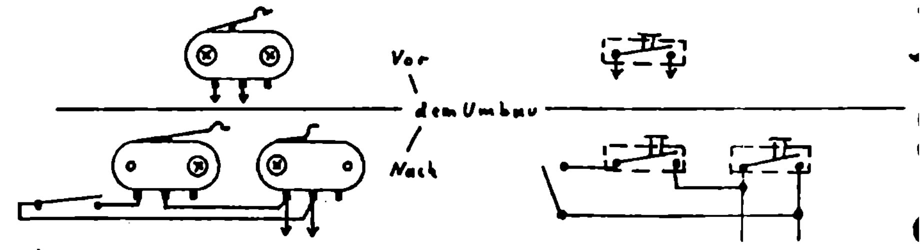
> Wenn der Druckkopf festhängt Drucker aus- und einschalten. Sollte dies nicht helfen,schaltet entweder einer der Mikroschalter nicht durch oder eine Verbindungsleitung wurde vergessen.

1.) 5 Gehäuseschrauben auf der Druckerunterseite lösen (eine in der Mitte,4 unter den Gummifüßen) und das Gehäuse vorsichtig öffnen.

2.) Anschlüsse der zum Druckwerk führenden Drähte notieren und diese von der Platine abziehen.

3.) 3 Befestigungsschrauben des Druckwerks lösen und es vorsichtig herausheben.Druckwerk so drehen,daß Mikroschalter und Motor nach oben weisen.Befestigungsschrauben des Mikroschalters lösen.

4.) Alten und neuen Mikroschalter nach Abbildung in den Löchern des alten Mikroschalters festschrauben.Elektrische Verbindungen herstellen.Kanten des Druckkopfes an den Stellen,wo er von den Fahnen der Mikroschalter berührt wird ein wenig abschrägen (Schlüsselfeile).



5.) Verbindungskabel des Druckwerks wieder mit der Platine verbinden,Schalter auf Normal (=Aus),Papierrolle einlegen und einlaufen lassen.

6.) Zunge des rechten Mikroschalters so verbiegen,daß er durchschaltet bevor der Druckkopf an der rechten Seite anschlägt. Probelauf mit wiederholtem "ADV".

7.) Schalter auf Graphik (=Ein),Zunge des linken Mikroschalters so verbiegen,daß Vorschub halbiert wird.Probelauf mit Druckermodus "MAN" und dem Programm FIX Ø ; Ø , LBL ØØ , PRX , GTOØØ. Stellen sich Unregelmäßigkeiten in den Zeilenabständen ein, muß die Zunge des Mikroschalters nachgebogen werden.Nicht die Nerven verlieren,es dauert etwas,bis man die richtige Ein-

stellung gefunden hat.

8.) Vorschubschalter an geeigneter Stelle am Gehäuse anbringen und beschriften.

9.) Druckwerk wieder einbauen,Gehäuse schließen,Probelauf wiederholen,fertig?-Hoffentlich!

Anwendung:

Wir haben nun 133 Spalten Schmalschrift = 19 Zeichen zur Verfügung.Mögliche Befehle in der Stellung Graphik sind: "ACCOL,BLDSPEC, ACSPEC,SKPCOL,SKPCHR,ACA,ACX" und zum Ausdruck der Zeile "ADV" (nicht "PRBUF"!).Es ist darauf zu achten,daß die oben angegebene maximale Spaltenzahl nicht überschritten wird.Alle anderen Druckerfunktionen sollten nur in der Stellung Normal ausgeführt werden.

Programme zum Bar-Code drucken:

Unterprogramm "BP"=Byte Print

Dieses Programm druckt das Byte,das in dezimaler Form im X-Register übergeben wird.Ferner werden in Speicher Ø3 alle Bytes addiert. Beim ersten Byte einer Zeile ist Flag ØØ zu setzen und Speicher Ø3 zu löschen.Beim letzten Byte einer Zeile ist nur Flag Ø1 zu setzen, dieses Byte enthält die Prüfsumme, die vom Programm automatisch gebildet wird.

Benutzte Speicher: ØØ + Ø3 Benutzte Flags: ØØ,Ø1

Size: ØØ4 , 174 Bytes

```
01◆LBL "BP"
02 .007
03 STO 00
04 RDN
05 FS?C 00
06 XEQ 15
07 ST+ 03
08 FC?C 01
09 GTO 00
10 ST- 03
11 RCL 03
12 256
13 /
14 INT
15 LASTX
16 FRC
17 256
18 *
19 +
20 XEQ 00
21 XEQ 18
22 XEQ 16
23 XEQ 18
24 XEQ 16
25 XEQ 17
26 ADV
27 ADV
28 ADV
29 ADV
30 ADV
31 RTN
32◆LBL 00
33 XEQ 18
34 XEQ 16
35 2
36 /
37 INT
38 LASTX
39 FRC
40 X≠0?
41 XEQ 16
42 X≠0?
43 XEQ 16
44 RDN
45 ISG 00
46 GTO 00
47 RTN
48◆LBL 16
49 2
50 RCL 01
51 INT
52 Y↑X
53 ST+ 02
54 RDN
55 ISG 01
56 RTN
57◆LBL 17
58 R↑
59 0
60 X<> 02
61 BLDSPEC
62 LASTX
63 BLDSPEC
64 LASTX
65 BLDSPEC
66 LASTX
67 BLDSPEC
68 LASTX
69 BLDSPEC
70 LASTX
71 BLDSPEC
72 LASTX
73 BLDSPEC
74 ACSPEC
75 ACSPEC
76 ACSPEC
77 ADV
78 .005
79 STO 01
80 RDN
81 RDN
82 RTN
83◆LBL 18
84 ISG 01
85 GTO 01
86 XEQ 17
87◆LBL 01
88 RCL 01
89 2
90 X>Y?
91 ISG 01
92 RDN
93 RDN
94 RTN
95◆LBL 15
96 ADV
97 SF 12
98 58
99 STO 02
100 RDN
101 XEQ 17
102 END
```

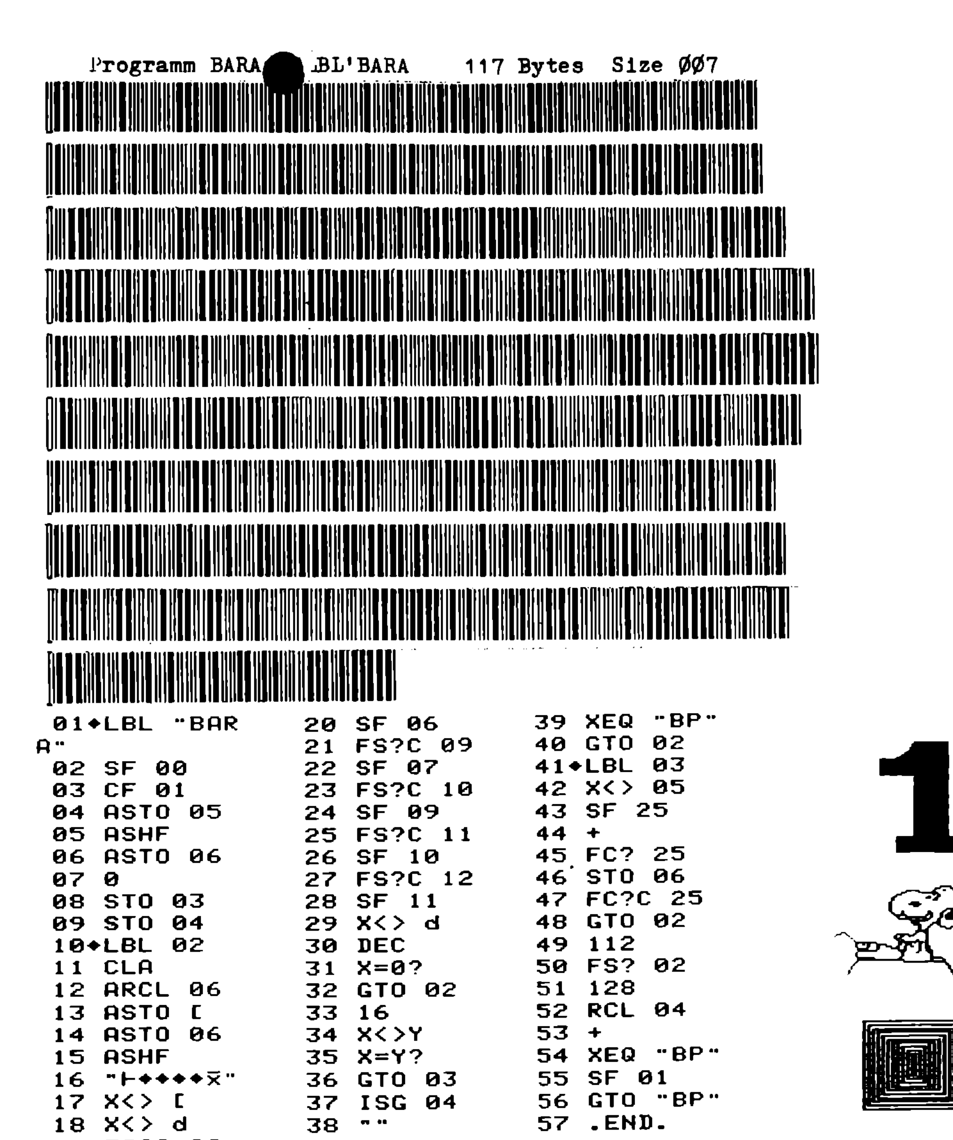
Programm "BARA"=Bar Alpha

Dieses Programm druckt Alpha-Data-Bar-Codes.Zeichenkette ins Alpha-Register eingeben,Bara ausführen,fertig.Als Unterprogramm wird "BP" benutzt.Die maximale Länge der Alpha-Kette beträgt 12 Zeichen.Wird vor dem Start Flag Ø2 gesetzt,wird ein "Append-Code" erzeugt,d.h. er wird beim Einlesen an das Alpha-Register angehängt.

Benutzte Speicher: ØØ - Ø6 Benutzte Flags: ØØ - Ø2

Size: ØØ7 , 119 Bytes

Programm BARA LBL'BARA   117 Bytes Size 007



```
01♦LBL "BAR      20 SF 06        39 XEQ "BP"
A"               21 FS?C 09      40 GTO 02
02 SF 00         22 SF 07        41♦LBL 03
03 CF 01         23 FS?C 10      42 X<> 05
04 ASTO 05       24 SF 09        43 SF 25
05 ASHF          25 FS?C 11      44 +
06 ASTO 06       26 SF 10        45 FC? 25
07 0             27 FS?C 12      46 STO 06
08 STO 03        28 SF 11        47 FC?C 25
09 STO 04        29 X<> d        48 GTO 02
10♦LBL 02        30 DEC          49 112
11 CLA           31 X=0?         50 FS? 02
12 ARCL 06       32 GTO 02       51 128
13 ASTO [        33 16           52 RCL 04
14 ASTO 06       34 X<>Y         53 +
15 ASHF          35 X=Y?         54 XEQ "BP"
16 "├♦♦♦♦♦x̄"    36 GTO 03       55 SF 01
17 X<> [         37 ISG 04       56 GTO "BP"
18 X<> d         38 ""           57 .END.
19 FS?C 08
```

1

Diese Seite bietet eine Kostprobe der selbstgedruckten Bar-Codes und der Graphik. Wer den Umbau nicht selbst durchführen möchte, der kann ihn* mir zusammen mit einem Scheck über 69,-DM zusenden. Ich kann jedoch keinerlei Garantie auf den Umbau oder daraus entstehende Folgen übernehmen. Mein Drucker arbeitet nach dem Umbau schon 3 Monate einwandfrei.

Winfried Maschke (413)
Eichhardtstr. 3
5276 Wiehl 1

Tschüss &



* Den Drucker



Tips und Tricks

Hallo Clubmitglieder !
Wie findet Ihr die Idee , dass regelmaessig im Prisma eine Seite mit Tips und Tricks erscheint ? !

Jedes Mitglied kann kurz und knapp Arbeitsanweisungen bzw. Erklaerungen zu speziellen Problemen in Form von Tips und Tricks geben , die nach geeigneter Durchsicht abgedruckt werden und in einer Ausgabe des Prismas erscheinen. Fehlerhafte und unvollstaendige Tips und Tricks werden nicht veroeffentlicht.

Ich habe mit Oliver ueber meine Idee gesprochen . Er fand das Ganze einen Versuch wert und bat mich , die Idee zu realisieren. Mit seinem Einverstaendniss bin ich also der Herausgeber der Tip- und Trickseite.

P.S. Ich wuensche mir viele Tips und Tricks.

Meine Adresse:
Walter Pieperhoff
Walb. Osth. Wallstr. 10
4770 Soest

Hier der erste Tip von Oliver !

Bei Ausfuehrung eines Programms in single step (Druck auf Taste SST) und angeschaltetem Drucker im Trace - Mode zeigt der Ausdruck immer eine Zeilennummer groesser an.
Erklaerung: Bei Druck auf Taste SST wird der Zeilenzaehler in Register e hochgezaehlt und beim Zugriff des Druckers ausgdruckt.

Nun ein Tip von mir:

Unterprogramm zur Eingabe von Zahlen .

Wenn grosse Mengen von Zahlen eingegeben werden muessen bietet sich folgendes Unterprogramm an .

```
LBL'U'  CLST  LAST X  X<> Z  LBL 01 R↓  1 ST+ T  R↑  STOP
STO IND T   DSE Z  GTO 01  RTN  o. END
```

Das Programm wird wie folgt aufgerufen :
z. B. 6 1/X XEQ 'U'
In diesem Beispiel fordert das Programm 6 Ziffern an die von Register 01 bis Reg. 07 abgespeichert werden.
Vorteilhaft wird die ganze Sache erst wenn das Unterprogramm von mehren Hauptprogrammen aufgerufen wird.

Tips und Tricks

Noch ein Tip von mir !

Erklaerung des Programmzeigers im b -Register

Nach RCL b steht folgendes in der Anzeige 0.0000 -61 .
Nach Ausfuehrung des Programms "DECODE" aus dem Buch von Wickes steht der Programmzeiger in Hexadezimal in der Anzeige (z.B. 00:00:00:00:00:31:4F).

Die letzten 3 Halbbytes (hier 14F) zeigen auf das Register 1 = 255 + 4F = 79 = 334 . Der Programmzeiger steht also im Register 334 auf dem 3.Byt.
Liste des 4. Halbbytes (nach Ausf. von DECODE)

0 = 1. Byt
1 = 2. Byt
2 = 3. Byt
3 = 4. Byt
4 = 5. Byt
5 = 6. Byt
6 = 7. Byt

(Achtung das Programm 'DECODE' ist von mir im Zuge des Magnetkarten-copyservice auf einer Karte zu erhalten.)

Der letzte Tip von mir !

Die Statuskarte als Programmanfangskarte !

Als erste Karte eines Programms kann man immer eine Statuskarte erstellen.
Die Vorteile :
Schaltet zum Beispiel den USER - Modus ein.
Dadurch koennen alle Tastenfeldzuordnungen mit eingelesen werden. (Vergessen ist menschlich)
Man braucht kein SIZE mehr auszufuehren. In der Anzeige kann sofort die erste Programminformation stehen.
Im Alpharegister steht die erste Druckzeile.
In den Registern Y ; Z ; T ; L koennen Werte oder Alphazeichen stehen ; die man sofort verwenden kann.

Als Nachteile sind anzufuehren :
Es muss immer eine Karte mehr eingelesen werden.
Beim zweiten Programmstart muss erst immer wieder die Statuskarte eingelesen werden um gleiche Anfangsbedingungen zu haben.

P R O G R A M M I E R H I L F E N   F Ü R   A N F Ä N G E R

Programmieren -und da bildet auch der HP-41 C keine Ausnahme- hat eine gewisse Ähnlichkeit mit dem Schachspiel: nach Durcharbeiten des Handbuchs kennt man zwar die "Spielregeln", aber Taktik und Strategie (Tricks im Kleinen und Programmablauf im Großen) bereiten oft noch Schwierigkeiten.

Hier soll nun die neue Rubrik "Programmierhilfen für Anfänger" Erfahrungen aus der Praxis für die Praxis vermitteln, die die Benutzung des HP-41 C wirklich zum "Happy Programming" machen. Daß hierbei evtl. auch der gehobene Anfänger oder gar der Fortgeschrittene profitiert, hofft der Verfasser, der sich allerdings nicht als Alleinunterhalter versteht: jedes Clubmitglied ist aufgerufen, Kritik, Anregungen und eigene Beiträge zu liefern. Sie werden dankend entgegengenommen von

Klaus Werner Hoenow
Alaskaweg 16
2000 Hamburg 73

Nun zum Inhaltlichen dieser neuen Rubrik:

Hier sollen keine längeren Programme für spezielle Problemstellungen gebracht werden, dafür ist anderswo im "prisma" ausreichend Platz. Kurze Programmstücke, ähnlich den "25 Words" der HP Key Notes, die besondere Programmiertricks enthalten, aber auch allgemeine Verfahrensbeschreibungen zur Speicherplatzeinsparung, zum sinnvollen Schleifenaufbau, zur Laufzeitverkürzung u. ä. sind gefragt. Hinweise zur Fehlersuche und -vermeidung, zur Programmkorrektur und -optimierung sollen gebracht werden; ferner ist an eine Einführung in den Umgang mit Flußdiagrammen sowie in die Strukturierte Programmierung gedacht.

Kurz und gut: alles was geeignet ist, den Weg von den Spielregeln zu Taktik und Strategie zu ebnen, soll in dieser Rubrik Platz finden; zur Mitarbeit sei hiermit jeder aufgerufen!

```
 01◆LBL "SAV
E"
 02 CLST
 03 FS? 39
 04 1
 05 FS? 38
 06 2
 07 FS? 37
 08 4
 09 FS? 36
 10 8
 11 +
 12 +
 13 +
 14 FS? 40
 15 .1
 16 FS? 41
 17 .2
 18 +
 19 STO 10
 20 END
```

```
 01◆LBL "DIS
P"
 02 PCL 10
 03 ENTER↑
 04 FRC
 05 10
 06 *
 07 GTO IND
X
 08◆LBL 00
 09 SCI IND
Y
 10 RTN
 11◆LBL 01
 12 FIX IND
Y
 13 RTN
 14◆LBL 02
 15 ENG IND
Y
 16 END
```

An den Anfang sei ein Optimierungsbeispiel gestellt:

Zu dem von mir in "prisma" 7-81 veröffentlichten Programm FORMAT gingen Verbesserungsvorschläge von Andreas Meyer-Lindenberg (258) und Holger Pochert (157) ein, von denen der letztere hier diskutiert werden soll, da er zwei für diese Rubrik geeignete Verfahren enthält.

Wie vielleicht erinnerlich, sollte anhand des Status der Flags 40 und 41 eine Kennzahl für das Anzeigeformat und anhand des Status der Flags 36 bis 39 die Anzahl der Nachkommastellen in der Anzeige (Bedienungshandbuch: Seite 231) bestimmt werden. Letzteres

wurde im Programm FORMAT mit einer DSE-Schleife bewerkstelligt, wobei die zum Bilden der Nachkommastellenzahl benötigten Summanden jeweils durch Verdopplung des Summanden aus dem vorhergehenden Schleifendurchlauf erzeugt wurde. Dieses Verfahren ist ebenso kompliziert wie dieser Text, der es beschreibt, was natürlich die Kritiker auf den Plan rufen mußte.

Wie man an den Zeilen 02 bis 13 des Programms SAVE von Holger Pochert sieht, läßt sich das Problem wesentlich eleganter ohne Schleife lösen, indem die Flags 39 bis 36 nacheinander direkt abgefragt werden und ggf. 1, 2, 4 und/oder 8 im X-Register aufsummiert wird. Andreas Meyer-Lindenberg schlägt das gleiche Verfahren vor.

Man sollte sich allerdings durch die bei diesem Beispiel ins Auge fallende Eleganz nicht dazu verleiten lassen, dieser direkten Methode in jedem Fall den Vorzug vor einer DSE- oder ISG-Schleife zu geben: Sobald nämlich die Anzahl der Flag- (oder sonstigen) Abfragen größer als 4 wird, kommt man evtl. bei einer Schleife mit weniger Programmspeicherplatz aus, auch wenn man dann gezwungen sein sollte -wie im Beispiel FORMAT- die aufzuaddierenden Summanden in jedem Schleifendurchlauf nach einer bestimmten Rechenvorschrift (hier: Verdopplung) neu zu bilden.

Der zweite Trick im Programm SAVE ist häufig anzutreffen, nämlich die Speicherung zweier ganzer Zahlen, getrennt durch Dezimalpunkt, in nur einem Register. Hier wurde die Anzahl der Nachkommastellen vor den Dezimalpunkt, die Kennziffer für das Format dahinter gesetzt und beides zusammen nach R10 abgespeichert (Zeilen 14 bis 19). Voraussetzung für diese Methode ist, daß man die maximal mögliche Stellenzahl der zweiten ganzen Zahl kennt, da diese sich sonst in die vor dem Dezimalpunkt stehende erste Zahl hinein ausbreiten könnte, und dann sind beide nicht mehr zu gebrauchen.

Eine mögliche Form der Auswertung eines solcherart gespeicherten Zahlenpaares zeigen die Zeilen 02 bis 06 des Programm DISP von Holger Pochert, das die Regenerierung des durch SAVE gespeicherten und im weiteren Gang eines anderen Programmes möglicherweise zerstörten Anzeigeformates bewirkt.

Allgemein läßt sich das Verfahren folgendermaßen beschreiben: Gegeben seien 2 ganze Zahlen K und L mit $L < 10^n$. Dabei soll n die kleinste ganze Zahl sein, die diese Ungleichung noch erfüllt (also z.B. n=2 für L maximal 99). Die Speicherung erfolgt, indem die Größe $K+L/10^n$ gebildet und nach R.. abgespeichert wird. Um K und L wieder getrennt zu erhalten, werden die Schritte RCL..; INT; LAST X; FRC; 1 E n; * durchgeführt. Dann steht K im Y- und L im X-Register. Wird im Folgenden (z. B. durch indirekte Operationen wie bei Zeilen 09, 12 und 15 von DISP) ohnehin nur der ganzzahlige Anteil der im Y-Register stehenden Zahl benutzt, so kann INT; LAST X durch ENTER↑ ersetzt werden.

Im Einzelfall ist allerdings zu überprüfen, ob die Einsparung an Datenspeichern nicht durch den Mehraufwand an Programmspeicherplatz zunichte gemacht wird.

Klaus Werner Hoenow

. Ordo
====

Spielprogramm für HP-41C ohne Erweiterungen
Andreas Meyer-Lindenberg (258), den 01.05.81

Anc... Meyer-Lindenberg
Elliger ... 31
5300 Bonn 2 - Bad Godesberg
Telefon 0228/323255

Size Ø13
Assignments (User): "ORDO"

Lieber Oliver, liebe Clubmitglieder,

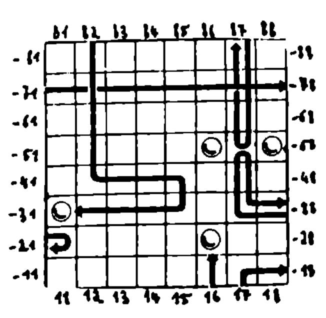
im Gegensatz etwa zu den USA ist das Spiel "Ordo" bei uns noch immer ziemlich unbekannt. Damit das nicht so bleibt, stelle ich euch hier eine "computer"isierte (HPisierte?) Version des Spiels vor und hoffe, daß das Spiel euch gefällt. Wie auch immer, hier sind die Spielregeln:

Ordo wird auf einem 8x8-Spielfeld wie dem unten gezeigten gespielt. Auf diesem Spielfeld versteckt der Rechner nun zu Beginn des Spiels eine bestimmt Anzahl von "Atomen", kleinen Kugeln. Die Positionen dieser Kugeln zu bestimmen, das "Molekül" zu analysieren, ist das Ziel des Spiels.

Sie erhalten jedoch vom Rechner keinerlei direkte Information über die Positionen der Atome (etwa: "A4"? "Versenkt"!). Sie können lediglich versuchen, aus dem Verhalten gewisser (imaginärer) Strahlen, die Sie von jedem Feld des Randes aus in das Spielfeld hineinschicken können, Ihre Schlüsse zu ziehen. Diese Strahlen werden nach gewissen einfachen logischen Regeln von den Atomen, abhängig von der Stellung, abgelenkt. Sie erhalten jeweils nur das Austrittsfeld des Strahls genannt. Ihre Aufgabe ist es nun, eine Molekülstellung zu finden, die zu dem Verhalten der Srahlen passt.

Für die Strahlenwege gelten die folgenden Regeln:

1. Strahlen bewegen sich immer gradlinig, nie diagonal! Befände sich kein Atom auf dem Spielfeld, so würden die Strahlen einfach auf der gegenüberliegenden Seite wieder austreten. Interessant wird es erst, wenn der Strahl von einem Atom beeinflußt wird:
2. Trifft ein Strahl genau frontal auf ein Atom, so wird er einfach verschluckt (absorbiert). Der Rechner gibt die Meldung *ABS* aus. Auf dem Beispielbrett geschieht solches dem Strahl von Feld 16.
3. Nähert sich ein Srahl einem Atom bis auf ein Feld (diagonal), so wird er rechtwinklig vom Atom weg abgelenkt (z.B. der Strahl von 17 auf dem Beispielbrett).
4. Kommt ein Strahl auf ein Feld genau zwischen zwei Atomen (wie der Strahl von Feld 87 im Beispiel), so wird er reflektiert (d.h er tritt an seinem Eintrittsfeld wieder aus). Der Rechner meldet *RFL*.
5. und letztens: tritt ein Strahl direkt neben einem Atom ins Spielfeld ein (wie der Strahl von Feld -21 im Beispiel), so wird er ebenfalls reflektiert.



Natürlich kann ein Strahl auf seinem Weg mehrmals abgelenkt werden. Der Strahl von Feld 82 z.B. wird dreimal nach Regel 3 abgelenkt, bevor er schließlich direkt auf ein Atom trifft und absorbiert wird. Wenn Sie nun glauben, die Stellung aufgeklärt zu haben, können Sie das Spiel stoppen und dem Rechner die Stellung eingeben. Stimmt Sie, so haben Sie das Spiel gewonnen. An-

dernfalls geht das Spiel wieder weiter.- bis Sie die Stellung enträtselt haben.

Je weniger Strahlen Sie zur Bestimmung eines Moleküls benötigen, desto besser sind natürlich Ihre analytischen Fähigkeiten. Um Ihre Spielstärke zu bewerten, verwendet der Rechner ein System von Minuspunkten wie folgt:

1 Minuspunkt .... wenn der Strahl absorbiert wird
2 Minuspunkte .... wenn der Strahl wieder austritt (also auch bei reflektierten Strahlen)
5 Minuspunkte .... wenn Sie eine Atomposition (am Ende) falsch geraten haben.

Je weniger Minuspunkte man am Ende gesammelt hat, desto besser natürlich. Man kann so auch gegen einen Freund im Wettkampf antreten - gewonnen hat der, der eine Stellung mit weniger Minuspunkten enträtselt hat.

Am Anfang ist ein Spiel mit vier Atomen zu empfehlen, danach kann man sich auf bis zu sieben Atome steigern. Soweit die Spielregeln - es folgt die Bedienungsanleitung:

1. Wenn Sie kein Erweiterungsmodul besitzen: Master-Clear SIZE Ø13. Da sämtliche Register belegt sind, können keine Assignments mehr gespeichert werden.
2. XEQ "ORDO". Der Rechner fragt nach einer Zufallszahl (RND?). Tippen Sie eine Zufallszahl zwischen Ø und 1 ein und drücken Sie R/S. Nun will der Rechner die Anzahl der Kugeln in der Stellung wissen (KG?). Geben Sie die Anzahl ein und drücken Sie R/S. Der Rechner erzeugt nun die zu ratende Stellung.
3. Jetzt beginnt das eigentliche Spiel: Der Rechner fragt nach dem Eintrittsfeld für einen Strahl (RAY?). Geben Sie die entsprechende Zahl ein (die Numerierung ersehen Sie aus dem Beispielbrett) und drücken Sie R/S. Nach etwas Rechenzeit erscheint nun entweder das Austrittsfeld (mit der Bezeichnung EXT:), oder die Meldungen *ABS* (= absorbiert) oder *RFL* (= reflektiert).
4. Nachdem Sie sich diese Meldungen auf dem Spielbrett notiert haben, drücken Sie R/S. Es erscheinen Ihre Minuspunkte (MIN:). Drücken Sie nochmals R/S, und es geht mit Schritt 3 weiter.
5. Wiederholen Sie die Schritte 3 und 4, bis Sie die Stellung komplett bestimmt zu haben glauben. Dann drücken Sie (mit USER-on) "B". Der Rechner fragt nach der Anzahl der Kugeln, die Sie raten wollen. Geben Sie sie ein und drücken Sie R/S.
6. Der Rechner fragt nach der n-ten zu ratenden Position (POS n?). Geben Sie die Position im Format Zeile,Spalte ein (die Atome auf dem Beispielbrett müßten etwa wie folgt eingegeben werden: 1,3 6,2 6,5 8,5) und drücken Sie R/S. War die Position richtig, so geht es mit der nächsten Position mit Schritt 6 weiter, bis die gesamte Stellung richtig eingegeben worden ist. Dann erscheint Ihr entgültiges Ergebnis (SCORE:), und Sie haben gewonnen. Wenn Sie aber ein Atom falsch raten ...
7. Erscheint die Meldung *ERR*, und es geht mit Schritt 4 weiter. Bedenken Sie aber, daß bereits richtig geratene Atome nicht mehr auf dem Brett sind!

So, das wär's. In der Hoffnung, daß euch das Spiel Spaß macht, und mit

HAPPY PLAYING
bin ich

Ordo

Spielprogramm für HP-41C ohne Erweiterungen
Andreas Meyer-Lindenberg (258), den 01.05.81

```
Ø1 LBL "ORDO"
Ø2 "RND?"          Zufallszahl
Ø3 PROMPT
Ø4 CLRG
Ø5 STO Ø9
Ø6 "KG?"           Anzahl
Ø7 PROMPT          Kugeln
Ø8 STO 1Ø
Ø9 LBL 1Ø          Zufallsposi-
1Ø 1               tion erzeugen
11 8
12 RCL Ø9
13 ACOS
14 FRC
15 STO Ø9
16 *
17 +
18 8
19 LAST X
2Ø FRC
21 *
22 INT
23 1Ø X
24 STO 11
25 RCL IND Y       Platz schon
26 *               besetzt ?
27 FRC
28 1Ø
29 ST* 11
3Ø *
31 INT
32 X≠Ø?            Ja, neuer Ver-
33 GTO 1Ø          such
34 RCL 11
35 1/X             Nein, Atom
36 ST+ IND Z       abspeichern
37 DSE 1Ø
38 GTO 1Ø          nächstes Atom
39 LBL Ø8
4Ø SF Ø5           Strahlwegbe-
41 SF 1Ø           rechnung
42 SF Ø9
43 11
44 STO Ø9
45 12
46 STO 1Ø
47 8Ø
48 "RAY?"          Eintritts-
49 BEEP            feld
5Ø PROMPT
51 X(=Ø?           Umformung in
52 GTO Ø1          Richtungsangaben
53 CF Ø9
54 X()Y?
55 CF 1Ø
56 X() Ø9
57 X() 1Ø
58 X() Ø9
59 LBL Ø1
6Ø ABS
61 1Ø
62 /
63 ENTER
64 INT
65 STO 12
66 -
67 1Ø
68 *
69 STO 11
7Ø FS? Ø9
71 8
72 X=Y?
73 CF 1Ø
74 LBL Ø6          Brett nach Atomen
75 CLST            absuchen, die den
76 XEQ Ø9          Strahl ablenken
77 X≠Ø?            können
78 GTO Ø2
79 1
8Ø ST+ IND 1Ø
81 XEQ Ø9
82 2
83 ST- IND 1Ø
84 XEQ Ø9
85 1
86 ST+ IND 1Ø
87 RDN
88 X=Ø?
89 GTO Ø4          "freie Fahrt"
9Ø FS? Ø5
91 GTO Ø3          Reflektionam Rand
92 1
93 FC? 1Ø
94 CHS
95 ST- IND Ø9
96 RDN
97 1Ø
98 X()Y
99 X)Y?
1ØØ GTO Ø3         Reflektion
```

```
101 FC?C Ø9      Ablenkung
102 SF Ø9
103 X<> Ø9
104 X<> 1Ø
105 X<> Ø9
106 FC? 1Ø
107 1
108 X≠Y?
109 GTO Ø4
11Ø FC?C 1Ø
111 SF 1Ø
112 LBL Ø4       Nächstes zu
113 CF Ø5        passierendes
114 1            Feld berech-
115 FC? 1Ø       nen ---
116 CHS
117 ST+ IND Ø9
118 8            Am Rand ange-
119 RCL IND Ø9   langt ?
12Ø X<=Ø?
121 GTO Ø5       Ja, zur Ausgabe
122 X<=Y?        Nein, nächstes
123 GTO Ø6       Feld
124 LBL Ø5
125 RCL Z        Austrittsfeld
126 ST- IND Ø9   konstruieren
127 RCL 12
128 1Ø
129 *
13Ø RCL 11
131 +
132 FS? Ø9
133 CHS
134 "EXT:"
135 ARCL X
136 2
137 LBL A        Meldung ausge-
138 ST- ØØ       ben und Minus-
139 BEEP         punkte berech-
14Ø PROMPT       nen
141 "MIN:"
142 ARCL ØØ
143 PROMPT
144 GTO Ø8       neuer Strahl
145 LBL Ø2       absorbiert
146 1
147 "*ABS*"
148 GTO A        ausgeben
149 LBL Ø3       reflektiert
15Ø 2
151 "*RFL*"
152 GTO A        ausgeben
153 LBL B        Positionstest und
154 "KG?"
155 PROMPT       Anzahl Kugeln
156 STO Ø9
157 LBL 15
158 "POS "
159 ARCL Ø9
16Ø " ?"         n-te Position
161 PROMPT       (x.y)
162 INT
163 STO 11
164 LAST X
165 FRC
166 1Ø
167 *
168 STO 12
169 CLST         befindet sich dort
17Ø XEQ Ø9       ein Atom?
171 X=Ø?         Nein, Spielerirr-
172 GTO 14       tum
173 TONE 8
174 RCL 11       Atom entfernen
175 1Ø X
176 1/X
177 ST- IND 12
178 DSE Ø9       nächstes Atom
179 GTO 15
18Ø "SCORE:"     alle Positionen
181 ARCL ØØ      korrekt, Anzeige
182 BEEP         Minuspunkte
183 PROMPT
184 LBL 14       Irrtumsroutine
185 5
186 "*ERR*"
187 GTO A        ausgeben
188 LBL Ø9       Unterprogramm; be-
189 RDN          rechnet Inhalt des
19Ø 1Ø           Feldes x.y (R12,R1
191 *
192 RCL 11
193 1
194 -
195 1Ø X
196 RCL IND 12
197 *
198 FRC
199 1Ø
2ØØ *
2Ø1 INT
2Ø2 +
2Ø3 END
```

© Andreas Meyer-Lindenberg
Elliger Höhe 31
5300 Bonn 2 - Bad Godesberg
Telefon 0228/323255

PROGRAM LISTING

PROGRAMMAUFLISTUNG
LISTAGE DU PROGRAMME
LISTATO DI PROGRAMMA

```
 01♦LBL "CRB
LD"
 02♦LBL A
 03 CLRG
 04 CF 21
 05♦LBL 00
 06 WNDSCN
 07 2
 08 STO 00
 09 X<=Y?
 10 GTO A
 11 RCL 01
 12 10
 13 MOD
 14 ST+ 00
 15 6
 16 X<Y?
 17 GTO 01
 18 RCL 01
 19 LASTX
 20 /
 21 INT
 22 X>Y?
 23 GTO 01
 24 10↑X
 25 RCL X
 26 RCL IND
00
 27 X<>Y
 28 /
 29 INT
 30 10
 31 MOD
 32 CHS
 33 X≠0?
 34 TONE 9
 35 SIGN
 36 *
 37 ST+ IND
00
 38 VIEW 01
 39 GTO 00
 40♦LBL 01
 41 2,008
 42 STO 01
 43 SF 21
 44♦LBL 02
 45 1
 46 STO 00
 47 CLST
 48 RCL IND
01
 49♦LBL 03
 50 RCL X
 51 10
 52 MOD
 53 -
```

```
 54 X<>Y
 55 LASTX
 56 RCL 00
 57 ST+ 00
 58 *
 59 +
 60 X<>Y
 61 10
 62 /
 63 X≠0?
 64 GTO 03
 65 X<> 09
 66 X<>Y
 67 BLDSPEC
 68 STO 09
 69 ISG 01
 70 GTO 02
 71 .END.
        GTO ..
PACKING
          CAT 1
LBL'CRBLD
END
112 BYTES
          SIZE 010
             FIX 2
             SF 27
        XEQ "CRBLD"
              0,00
             10,00
             20,00
             30,00
             40,00
             50,00
             60,00
             11,00
             22,00
             33,00
             44,00
             55,00
             66,00
             56,00
             46,00
             36,00
             26,00
             16,00
              6,00
             15,00
             24,00
             42,00
             51,00

"→ΩAβΩ⊦"      ***
             SF 12
        ACSPEC
⋈
```

```
        XEQ "CRBLD"
              0,00
              1,00
              2,00
              3,06
              4,00
              5,00
              6,06
              5,06   Piep!
              5,00
             11,00
             22,00
             15,00
             24,00
             33,00
             42,00
             44,00
             51,00
             55,00
             60,00
             61,00
             62,00
             63,00
             64,00
             65,00
             66,00

 "↓≠Y5◂A"     ***
             SF 12
        ACSPEC
⧗
             CLX
```

Korrektur eines falsch gelesenen Barcodes (5,06 / 5,00)

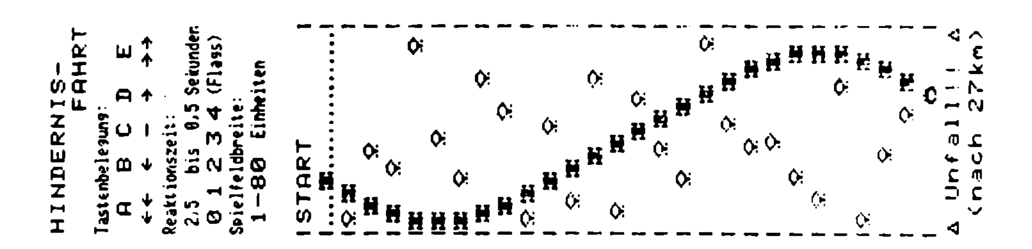
Ralf Wirowski

Arthur-Hantzsch-Str.12
5ooo Köln 8o
Tel: (o221) 663477

+++ mitgliedsnummer 359

Liebe Clubfreunde !

Seit längerer Zeit beschäftige ich mich mit einer Verwirklichung bekannter Telespiele auf dem 41er-System. Da hier kein Bildschirm vorhanden ist (wie lange noch?), muß zwangsweise der Drucker mit seiner hochauflösenden Graphik (sogar besser als Bauklotzgraphik bei TRS 8o) zu Aufgaben herangezogen werden. Diese Tage konnte ich ein „Hindernis-Fahrt"-Spiel fertigstellen. Hier nun das Resultat :



Durch konsequente Programmspeicherplatzeinsparung gelang es mir dieses Programm auf 3 Magnetkarten zu pressen. Leider führte dies zu einer Anwendung der synthetischen Programmierung. Zusätzlich habe ich in diesem Programm den vollen ASCII-Zeichensatz des Rechners angewendet, was mit einigen synthetischen Programmierkenntnissen aber ein Kinderspiel ist.

++++++ Dieses Programm möchte ich einem Copyright unterziehen. - Verbreitung also nur unter uns Clubmitgliedern! ++++++

mit freundlichen Grüßen

HINDERNIS-FAHRT

DIESES PROGRAMM STELLT DIE HOCHAUFLÖSENDE GRAPHIK DES 41c-DRUCKERS DURCH EIN SPIEL VOR.
DER SPIELER FÄHRT MIT EINEM AUTO EINEN WEG AB, AUF DEM SICH IHM STÄNDIG HINDERNISSE (BÄUME) IN DEN WEG STELLEN!!!
DIE REAKTIONSZEIT VERKÜRZT SICH AUTOMATISCH JE GEFAHRENEN 5KM.

ERFORDERLICHE EINGABEN:
KURSÄNDERUNG DURCH TASTENDRUCK BEI DER LAGEANZEIGE VON AUTO (A) UND HINDERNIS (H) INNERHALB DER REAKTIONSZEIT. OHNE TASTENDRUCK GEHT ES GERADEAUS WEITER.

```
 01♦LBL "H-F"
 02 F7000804
    38008001
 03 RCL [
 04 STO d
 05 F710E374
    48F13738
 06 RCL [
 07 STO 01
 08 F711DD17
    FFFFD177
 09 RCL [
 10 STO 02
 11 F7101251
    1E246504
 12 RCL [
 13 STO 03
 14 17
 15 STO 04
 16 "HINDERNIS-"
 17 PRA
 18 "FAHRT"
 19 ACA
 20 ADV
 21 CF 12
 22 "Tastenbelegung:"
 23 PRA
 24 SF 12
 25 1
 26 STO 00
 27 " A "
 28 XEQ 04
 29 "B C D "
 30 XEQ 04
 31 "E "
 32 XEQ 04
 33 "←← ← → →→"
 34 XEQ 04
 35 ADV
 36 CF 12
 37 "Reaktionszeit:"
 38 PRA
 39 " 2,5 bis 0,5"
 40 "⊢ Sekunden"
 41 PRA
 42 SKPCHR
 43 SF 12
 44 "0"
 45 XEQ 04
 46 "1"
 47 XEQ 04
 48 "2"
 49 XEQ 04
 50 "3"
 51 XEQ 04
 52 "4"
 53 ACA
 54 CF 12
 55 " (Flags)"
 56 ACA
 57 "Spielfeldbreite"
 58 "⊢:"
 59 PRA
 60 SKPCHR
 61 SF 12
 62 "1-80 "
 63 ACA
 64 CF 12
 65 "Einheiten"
 66 ACA
 67 PRBUF
 68 SF 12
 69 ADV
 70 XEQ 07
 71 3
 72 SKPCOL
 73 "START"
 74 ACA
 75 44
 76 SKPCOL
 77 XEQ 07
 78 ADV
 79 XEQ 07
 80 2
 81 SKPCOL
 82 CF 12
 83 "....."
 84 ACA
 85 SF 12
 86 1
 87 SKPCOL
 88 RCL 02
 89 ACSPEC
 90 1
 91 SKPCOL
 92 CF 12
 93 ACA
 94 ACA
 95 ACA
 96 ACA
 97 SF 12
 98 SKPCOL
 99 XEQ 07
100 GTO 06
101♦LBL 04
102 ACA
103 CF 12
104 SKPCHR
105 SF 12
106 RTN
107♦LBL 05
108 ARCL X
109 "⊢-"
110 6
111 +
112 ARCL X
113 RTN
114♦LBL 06
115 RCL 04
116 "A:"
117 XEQ 05
118 "⊢H:"
119 RCL 06
120 9821
121 *
122 .211327
123 +
124 FRC
125 STO 06
126 72
127 *
128 INT
129 1
130 +
131 STO 05
132 XEQ 05
133 4
134 0
135 ADV
136 SF 00
137 CF 21
138♦LBL 00
```

```
139 FS? IND Y
140 AVIEW
141 FS? IND Y
142 GTO IND Y
143 TONE IND Y
144 DSE Y
145 GTO 00
146 AVIEW
147 PSE
148◆LBL 01
149 PSE
150◆LBL 02
151 PSE
152◆LBL 03
153 PSE
154◆LBL 04
155 PSE
156 CLD
157 GTO C
158◆LBL A
159 -2
160◆LBL B
161 3
162 -
163 GTO C
164◆LBL E
165 2
166◆LBL D
167 3
168 +
169◆LBL C
170 SF 21
171 RCL 04
172 +
173 STO 04
174 X<=0?
175 GTO 09
176 75
177 X<=Y?
178 GTO 10
179 RCL 05
180 7
181 +
182 RCL 04
183 X>Y?
184 SF 05
185 RCL 05
186 -
187 ABS
188 8
189 -
190 X<0?
191 GTO 08
192 XEQ 07
193 RCL 04
194 RCL 05
195 FS? 05
196 X<>Y
197 STO 05
198 X<>Y
199 STO 04
200 SKPCOL
201 RCL 02
202 FS? 05
203 RCL 03
204 ACSPEC
205 RCL 04
206 7
207 +
208 CHS
209 RCL 05
210 +
211 SKPCOL
212 RCL 02
213 FC? 05
214 RCL 03
215 ACSPEC
216 75
217 RCL 05
218 -
219 SKPCOL
220 XEQ 07
221 RCL 05
222 FS?C 05
223 STO 04
224 4
225 RCL 00
226 5
227 /
228 INT
229 X<=Y?
230 SF IND X
231 1
232 ST+ 00
233 GTO 06
234◆LBL 07
235 127
236 ACCOL
237 RTN
238◆LBL 08
239 XEQ 07
240 RCL 04
241 SKPCOL
242 RCL 01
243 ACSPEC
244 75
245 RCL Z
246 -
247 SKPCOL
248 XEQ 07
249 GTO 11
250◆LBL 09
251 RCL 01
252 ACSPEC
253 76
254 SKPCOL
255 XEQ 07
256 GTO 11
257◆LBL 10
258 XEQ 07
259 76
260 SKPCOL
261 RCL 01
262 ACSPEC
263◆LBL 11
264 ADV
265 TONE 7
266 TONE 7
267 TONE 7
268 "Δ Unfall!! Δ"
269 PRA
270 "(nach "
271 ARCL 00
272 "├km)"
273 ACA
274 SF 29
275 FIX 4
276 8
277◆LBL 12
278 ADV
279 DSE X
280 GTO 12
281 END
```


R. Wi
Arthur-Hant
5000 Kö
Telefon: (02

SYSTEM:
HP-41c
+ DRUCKER
+ SPEICHER MODUL

SIZE 007 / 656 BYTES

HINDERNIS-FAHRT

Tastenbelegung:

A	B	C	D	E
←←	←	–	→	→→

Reaktionszeit
2,5 bis 0,5 Sekunden
0 1 2 3 4 (Flags)

Spielfeldbreite:
1–80 Einheiten

```
|START              |
|...................|
Δ Unfall!! Δ
  (nach 6km)
```

Frank Al■■sen

Towers of Hanoi HP-41 C

Dieses Programm bearbeitet eine leicht eingeschränkte - wohl aber die populärste Art - des Spiels TÜRME VON HANOI; nämlich die Version mit drei Türmen. Die Anzahl der Scheiben ist nur durch die Kapazität des Rechners begrenzt.
Der hier verwendete Algorithmus gibt als Ergebnis nur die beiden Türme an, die am nächten Umlegevorgang beteiligt sind, nicht aber die Umlegerichtung! Diese ergibt sich aus den Spielregeln.

Programmstart:	XEQ'HANOI		SCHEIBEN ?
Eingabe der Anzahl der Scheiben:	n	R/S	SCHEIBEN:N
			ZUGZAHL: M
			" Zug a "
			" Zug b "
			.
			.
			.
			.
			+++READY+++

SPIELREGELN : Es gilt, alle Scheiben des ersten Turmes, unter zuhilfenahme des zweiten, in der gleichen Reihenfolge auf dem dritten Turm aufzustapeln.
Es darf immer nur eine kleinere Scheibe auf eine größere zu liegen kommen! Wird also angezeigt: A CHANGE B, so heißt das, daß die kleinere der obersten Scheiben der Türme A und B umgelegt werden muß, und zwar auf den anderen der beiden Türme.

```
001 LBL'HANOI        021 STO 03        041 2
002 FIX 0            022 'ZUEGE:       042 RCL 02
003 'SCHEIBEN ?      023 AVIEW         043 X=Y ?
004 PROMPT           024 PSE           044 GTO 02
005 STO 00           025 LBL 00        045 '_2-CHANGE-3
006 'SCHEIBEN:       026 1             046 LBL 04
007 ARCL 00          027 ST+ 03        047 PROMPT
008 AVIEW            028 RCL 03        048 DSE 01
009 2                029 ENTER         049 GTO 00
000 RCL 00           030 ENTER         050 '++++READY+++
011 Y↑X              031 3             051 PROMPT
012 RND              032 /             052 GTO 04
013 1                033 INT           053 LBL 01
014 -                034 3             054 _1-CHANGE-3
015 STO 01           035 x             055 GTO 04
016 'ZUGZAHL:        036 -             056 LBL 02
017 ARCL 01          037 STO 02        057 '_1-CHANGE-2
018 AVIEW            038 1             058 GTO 04
019 PSE              039 X=Y ?         059 -.END.-
020 CLX              040 GTO 01
```

Datenspeicher: 00 Anzahl Scheiben
01 Anzahl Züge
02 Belegt
03 Zähler

Numerische Labels: 00, 01, 02, 04
Alpha-Labels : 'Hanoi'
SIZE: 004
Keine Erweiterungen nötig.

Die Anzeige der Programmbeschreibung erhält man ohne "R/S", wenn man den Programmschritt 047 durch AVIEW ersetzt.

Clubprojekte Nr.1 (Magnetkarten,-Taschen,-Hüllen) u. Nr.3 (Papier)
----

z. Hd.: Ulli Davertzhofen, Ostenbergstr. 97/8, 4600 Dortmund 50
Postscheckkonto-Nr. 2080 27-432, PschA Essen

Liebe Clubmitglieder,

wir haben uns entschlossen, das Clubprojekt Nr.3 (Thermopapier) zu beenden. Der Preisvorteil beim Einkauf größerer Mengen Papier ist mittlerweile sehr gering geworden, so daß Sammelbestellungen, die zudem noch einigen Arbeitsaufwand erfordern, kaum noch lohnen. Damit Ihr auch künftig Druckerpapier zu vernünftigen Preisen bekommt, möchte ich Euch bitten, direkt mit

Herrn Karl-Josef Meusch
Friedrich-Ebert-Straße 46
5020 Frechen-Königsdorf
Tel. 02234 / 84127

Kontakt aufzunehmen. Über Herrn Meusch habe ich auch alle bisherigen Sammelbestellungen abgewickelt und kann ihn deshalb nur empfehlen. Derzeit (Stand: 23.7.81) gelten folgende Preise:

HP-Papier, 57 mm, 25 m, Druck schwarz :

1 - 23 Rollen : DM 1.70 /Stück
24 - 50 Rollen : DM 1.61 /Stück
51 -101 Rollen : DM 1.56 /Stück (Preise inkl. Must. zzgl. Versand).

Natürlich werden wir auch künftig versuchen, beim Thermopapier "am Ball" zu bleiben, indem wir nach günstigen Angeboten Ausschau halten und Euch darüber informieren.

An: Ulli Davertzhofen, Ostenbergstr. 97/8, 46 Dortmund 50

Name: ______________________ Mitgl.-Nr.: __________

Anschrift: ______________ ,( ) ______________

Ich bestelle:

_____ mal 40 Magnetleerkarten , je DM 18.00 = DM

_____ Stück Kartentaschen(Spiralheftung), je DM 9.00 = DM

_____ Stück Einlageblätter(Klarsichtfolie,DIN A4, für je 15 Karten) , je DM 1.80 = DM

+ pauschaler Unkostenbeitrag(Verpackung,Porto) = DM 4.00

Gesamtbetrag = DM

==========

( ) Ich habe den Betrag auf Pschk 208027-432, Essen überwiesen.

( ) Verrechnungsscheck liegt bei.

P.S. Der Absender der DM 168,- (s. "prisma", S. 194/95-81) hat sich immer noch nicht gemeldet. Ich möchte ihn noch einmal bitten, sich umgehend bei mir zu melden! Bestellungen ohne Angabe des Auftraggebers und des Bestellumfangs kann ich nicht bearbeiten.

Lösungsbuch Nr. 3 - Intelligente Spiele

Das in "prisma" (S.65-81) angekündigte Lösungsbuch mit intelligenten Spielen ist fertig. Es enthält auf 35 Seiten fünf interessante Programme,u.z.:

1. Superhirn(verbesserte Fassung)- Finden Sie den geheimen Code , den der Rechner "versteckt" hat, in möglichst wenigen Versuchen (Anzahl Positionen und Farben frei wählbar; Farben einfach/doppelt).
2. Invers-Superhirn(verbesserte F.)- Drehen Sie den Spieß um und lassen den Rechner Ihre Kombination entschlüsseln. Er schafft das überraschend gut (Anzahl Positionen,Farben frei wählbar).
3. U-Boot-Jagd (verbesserte F.)- Wer findet und zerstört die fünf feindlichen U-Boote zuerst, Sie oder der HP-41 ?
4. Reversi- Ein strategisches Brettspiel gegen den Rechner
5. 4x4x4-TicTacToe- Versuchen Sie, auf einem dreidimensionalen Spielbrett eine Reihe mit vier Steinen zu bilden, bevor es dem Rechner gelingt. Das Programm ist dem im HP-Lösungsbuch "Games" veröffentlichten Pr. klar überlegen.

Lieferumfang:
Die Programme sind sauber dokumentiert und mit allen nötigen Erläuterungen versehen. Das Ganze wird mit Plastikhefter in einer stabilen Versandtasche verschickt. Auf Wunsch werden 16 beschriebene Magnetkarten (+ Einlegeblatt) beigefügt. Lieferung erfolgt gegen Vorkasse unmittelbar nach Zahlungseingang.

P.S. Eine Fortsetzung mit weiteren intelligenten Spielen ist geplant.

An: Ulli Davertzhofen, Ostenbergstr. 97/8, 46 Dortmund 50

Name: ____________________

Anschrift: ____________________,( ) ____________

Ich bestelle:

____ mal Lösungsbuch "Intell. Spiele" , ohne Magnetkarten , je DM 18.90 = DM

____ mal Lösungsbuch "Intell. Spiele" , mit 16 beschriebenen Magnetkarten + Einlegeblatt , je DM 28.90 = DM

Gesamtbetrag = DM

(Preise inkl. aller Nebenkosten)

( ) Ich habe den Betrag auf Pschk 2080 27- 432, PschA Essen überwiesen.
( ) V- Scheck liegt bei.

Und hier das neueste aus CHIP 8-81:

SEX-Versand tauscht Filme Super 8 und Video gegen Z-8o Software. Riesenauswahl.

Man muß die Marktlücke nur finden; oder die richtige Werbung betreiben. Wie hieß doch gleich ? Waschmaschine, Dolby-Rauschunterdrückung, 3o - 16o[illegible] Hz, Gleichlaufschwankungen: gering, 1 Jahr Garantie, Tel......

S T I M M Z E T T E L Z U R W A H L  1

Eine Honorierung von Beiträgen zu prisma erfolgt bekanntlich nicht. Um dennoch einen gewissen Anreiz zu bieten, erfolgt jährlich eine Wahl des besten Beitrages. Bitte überlegt bei der Auswahl des besten Berichts genau und seht Euch alle bisherigen Infos noch einmal an. Sie alle stehen zur Auswahl !
Auch Austritt aus dem Club und Meinung über den Mitgliedsbeitrag können verbindlich mitgeteilt werden. Der Austritt erlangt nur dann Geltung, wenn eine entsprechende Bestätigung bis zum 15.November erfolgte ! Andernfalls sicherheitshalber erneut kündigen.

Einsendeschluß des Bogens: 15. Oktober 1981
Bitte einsenden an: Olaf Gursch (120) 
Grüntenstr. 29 b
1ooo Berlin 42

Jedes Mitglied darf nur einen Stimmzettel abgeben; Selbstwahl eigener Beiträge ist ausgeschlossen. Wahl unter Ausschluß des Rechtsweges !

Am besten hat mir in allen Clubinfos bisher gefallen:

______________ ______________ ______
Artikel / Bericht / Programm etc. Autor Seite

Weiterhin haben mir sehr gut gefallen:

______________ ______________ ______

______________ ______________ ______

______________ ______________ ______

Folgende Themen interessieren mich besonders stark: (bitte einkreisen)

Hardware Synthetisches Spiele Programmierhilfen
Eletrotechnik Steuer und Finanzen Statik Statistik
Mathematik Sonstiges: ....

Bemerkungen (Stichwort !) :

Zutreffendes bitte ankreuzen :

☐ Ich habe Interesse an Clubtreffen in der Umgebung meines Wohnortes.
☐ Ich bin bereit diese Treffs zu organisieren
☐ Ich war schon bei einem Clubtreffen, außer auf der Hannovermesse !

☐ Ich bin für Einschränkung der Clubleistungen bei Beibehaltung des Jahresbeitrags
☐ Ich bin für Beibehalt der Clubleistungen bei Jahresbeitrag von 60,--DM für 1982
☐ Ich kündige hiermit meine Mitgliedschaft zum 1.1.82: Unterschrift: ______
☐ Ich kündige hiermit, falls der Beitrag auf 60,--DM für 1982 festgesetzt wird. Bitte Unterschrift: ....
☐ Bei 60,--DM Jahresbeitrag möchte ich diesen in zwei Raten zahlen können !

Weitere Bemerkungen, Anregungen etc. zusammen mit Name, Anschrift und Mitg.Nr. auf die Rückseite dieses Blattes. Nicht vergessen: Ab bis zum 15. Oktober !!!

Preise: Erststimmen zählen zwei Punkte, 2., 3., 4.-Stimmen je einen Punkt. Der Beitrag, der die meisten Stimmen erhält, wird mit einem ROM eigener Wahl belohnt. Ebenso der Autor, der für alle seine Programme insgesamt die meisten Stimmen (Pkt.) erhält.
Die monatlichen, allgemeinen Mitteilungen sind nicht wählbar ! 8 Trostpreise !!

Fragebogen zur Mitgliederliste, Blatt 1 (Beiblatt)

Der neu herausgegebene Fragebogen zur Mitgliederliste dient der Neuordnung der inzwischen kaum noch überschaubaren Mitgliederschaft unseres Clubs. Aus verständlichen Gründen ist es unmöglich, eine ständig aktuelle Liste der Clubmitglieder zu veröffentlichen. Darum wurde nun beschlossen, daß die Mitglieder des Clubs sowie wichtige Daten per Computer gespeichert und verwaltet werden. Genauer: Alle Daten, die bei mir zur Mitgliederkartei eintreffen,werden auf Disketten mithilfe eines Apple-II-plus gespeichert.

Natürlich kosten die Disketten Geld. Damit nun der Club nicht ganz arm wird, soll mit dem Apple ein Info-Dienst eingerichtet werden, der folgende Fragen beantworten kann (oder ähnliche):

z.B.: "Adresse von Herrn XY"
"Name von Mitglied Nr. xyz"
"Alle Mitglieder im Postleitzahlbereich 24xx"
"Alle Mitglieder im Telefon-Vorwahlbereich ....."
"Alle nicht volljährigen Mitglieder"
"Alle Drucker-Besitzer"
"Alle PPC-Modul-Besitzer"
"Alle Tennis-Spieler"
"Alle, die mir Programme auflisten"
etc., etc., etc.

Preis für solche Auskünfte: 40 DPf pro erhaltene Adresse, jedoch mindestens 3 DM.
(Bitte einen frankierten Umschlag beilegen)

Natürlich kann Ihnen niemand vorschreiben, die Fragebogen überhaupt einzuschicken. Falls Sie dies aber (hoffentlich) tun, bleiben Ihnen noch verschiedene Möglichkeiten für Ihre Eintragung in die Kartei:

Jeder normal eingehende Fragebogen wird gespeichert und dann in einem Ringordner abgeheftet. Die Daten stehen für Anfragen zur Verfügung.

Wenn Sie z.B. Ihre Telefon-Nr. nicht veröffentlicht haben möchten, so kreuzen Sie die Ziffer o5. mit einem roten Stift an. Alles, was mit rotem Stift geschrieben, angekreuzt oder durchgestrichen wurde, wird nur abgeheftet, aber nicht gespeichert!!!

Wenn Sie die Mitgliedsnummer rot schreiben, oder die Ziffer o1. rot ankreuzen, werden Sie im Computer überhaupt nicht registriert!

Alle schon registrierten Mitglieder (also die, die schon eine Mitgliedsnr. haben) schicken die Fragebogen gleich an die Kartei, Mitglieder, die noch keine Nummer haben, schicken den Bogen ausgefüllt an Herrn Rietschel, der ihn dann mit Nummer an mich weiterleitet.

Fragebogen zur Mitgliederliste, Blatt 2

o1. Mitgliedsnummer ____________

o2. Name ____________________________

Vorname ____________________________

o3. Straße ____________________________

Hausnummer ____________

evtl. Postfach ____________

o4. Postleitzahl ____________

evtl. Ortsteil ____________________________

Ort ____________________________

o5. Telefon-Vorwahl ____________

Rufnummer ____________

o6. Geburtsdatum ____________

o7. Hardware

() HP-41c
() HP-41cv
() Kartenleser
() Drucker
() Barcodeleser
() __ einfache Memorys (SDM)
() __ DDMs
() QDM
() Module im HP-41c
() Kassetten-Interface
() anderes: ____________________________

o8. Software

a) Module

() Mathe
() Finanz
() Statistik
() Spiele
() PPC
() Surveying
() Circuit Analysis
() Real Estate
() Navigation
() Aviation
() Thermal Science
() Machine Design
() Home Management
() Civil Engineering
() Clinical Lab
() Securities
() Stress Analysis
() anderes: ____________________________

b) HP-Solutions

() Test Statistik
() Maschinenbau
() Mathematik
() Elektrotechnik
() Business
() Home Construction
() Real Estate
() Lending, Saving, Leasing
() Small Business
() Geometry
() Antennas
() Chemical Engineering
() Civil Engineering
() Control Systems
() Electrical Eng.
() Fluid Dynamics
() Heating, Vent.
() Mechanical Eng.
() Solar Eng.
() Calendars
() Cardiac/Pulm.
() Chemistry
() Games
() Optometry I
() Optometry II
() Surveying
() Physics
() anderes: ____________________________

Fragebogen z[illegible]Mitgliederliste, Blatt 3

o9. Beruf: ______________________

1o. Hobbys:

() Jogging
() Radfahren
() Handball
() Volleyball
() Judo
() andere Budo
() Badminton
() Wasserball
() Ski alpin
() Leichtathlethik
() Fußball
() Basketball
() andere Ballspiele
() Karate
() Tennis
() Schwimmen
() andere Wassersport
() Ski nordisch
() anderes: ______________________

() Klassik
() dt. Schlager
() Volksmusik
() Jazz
() Rock
() Disco
() exp. Musik
() anderes: ______________________

() Instrument: ______________________

() Lesen: ______________________

() andere Computer: ______________________

() Elektronik: ______________________

() anderes: ______________________

11. Erstelle gegen Umkosten erstattung Listings von Magnetkarten ()

12. Plotte Barcodes ()

13. Datum, Unterschrift: ______________________

(Bitte lesen Sie auch das Beiblatt durch!!)

Lösungsbuch Nr. 2 im Druck

Das LöBu 2, für das inzwischen 19 Bestellungen vorliegen, geht in Kürze zum Druck. Der Preis beträgt wegen der relativ geringen Nachfrage 5o,--DM. Die Auflage wird auf einmalig 1oo Ex. limitiert. Ein Nachdruck erfolgt nicht. Werden 25 Ex. verkauft, so sind die Kosten gedeckt; der Erlös aus dem Verkauf weiterer Ausgaben wird dann - sofern mehr als 25 Stck. verkauft werden - auf die Autoren verteilt. Die Prgr. sind auf Magnetkarte erhältlich. Das LöBu 2 wurde um ein exklusives Biorhytmusplottprgr. erweitert, das aus 4 vorliegenden Versionen ausgewählt wurde und von Robert Klauo geschrieben wurde. Ferner wurde ein gutes Kalender-Plott-Prgr. von Manfred Mickoteit mit aufgenommen. Das LöBu ist ein Vielfaches seines Kaufpreises wert ! Bestellungen bitte mit V-Scheck oder Schein an Oliver (1).

HEWLETT-PACKARD ANWENDER-CLUB
OLIVER RIETSCHEL

ANFORDERUNGSSCHEIN FUER KARTENKOPIEN

Datum : ________________

Hallo ________________

Bitte sende mir folgende Programme :

Nr.	Name	Karten
1.		
2.		
3.		
4.		
5.		
6.		
7.		
8.		
9.		
10.		
11.		
12.		
13.		
14.		
15.		
16.		
17.		
18.		

Bitte Formular mit Anzahl der Karten u. frankierten adressierten Rueckumschlag zu den beiden Kartencopy's senden.Vielen Dank !

Beiliegend : ____________ Karten

Liebe Mitglieder !

Die heutige Oktoberausgabe werde ich bereits am 22. August fertigstellen, denn das derzeitige Versandverfahren zwingt zu solch langen Vorplanungen. Doch das wird ab dem nächsten Jahr ganz anders werden; bitte lest alle den Bericht "Ausblick 82" gründlich durch ! Auch in der Clubbibliothek gibt es - ab sofort - eine Neuerung: Bei der Anforderung von Büchern braucht keine Versandtasche mehr mitgeschickt werden ! Näheres s.u. ! Doch die wirklich großen Änderungen werden sich erst zum Jahreswechsel auftun. Siehe "Ausblick 82". Euer Oliver (1)

- Ich danke den vielen Zuschriften, die sich mit dem Thema "Büchersendung" und der Portoerhöhung beschäftigten. Die überwiegende Mehrheit stimmte meinem Bericht zu diesem Thema im letzten Info uneingeschränkt zu. Und so endeten auch fast alle Briefe mit dem zähneknirschenden Postulat: Wir müssen es doch wohl hinnehmen. Aber, liebe Mitglieder, dann kam Rolf mit der verheißungsvollen Mitgliedsnummer 5oo und sorgte dafür, daß der ganze "Organisationsapparat" unseres Clubs kräftig ins rotieren kam: prisma wird 1982 voraussichtlich gesetzt und trotz der höheren Setzkosten bei einem ca. Beitrag von 6o,--DM (keinesfalls mehr) monatlich erscheinen. Mehr unter "Ausblick 82"

- Bei Einsendung von Programmen an einen Projektleiter oder mich bitte neben Magnetkarten auch ein Beispiel beifügen ! Häufig stellte sich in der Vergangenheit nämlich heraus, daß wir falsche Programme bekamen. Manchmal ist dann sogar ein Fehlerchen mit ins prisma gekommen. Solltet Ihr aber Programme aus der Magnetkartenbibliothek anfordern, so könnt Ihr gewiß sein, immer die neuesten Versionen zu erhalten, bei denen veröffentlichte Fehler bereits ausgemerzt sind. Solltet Ihr in einem veröffentlichten Programm einen offensichtlichen Fehler gefunden haben, so teilt das bitte dem jeweiligen Autor mit. Diesen möchte ich dann bitten, den Fehler so bald wie möglich zu eliminieren und die verbesserte Version an die Magnetkartenbibliothek zu Walter oder Ralf zu schicken.

- Lösungsbuch Nr. 2 - Kalenderrechnungen - : Ich möchte auf diesem Wege den Autoren den Dank der vielen Zuschriften weitervermitteln, die dieses ausgezeichnete Werk sehr lobten. Um so bedauerlich ist, daß bis heute nur 18 Exemplare verkauft sind, was noch nicht einmal die Unkosten deckt. Eine Ergänzung zu dem LöBu 2 erscheint noch mit diesem Info. Wer das LöBu 2 haben möchte, erhält es sofort gegen 5o,--DM bar oder V-Scheck von Oliver (1). Sehr bedauerlich ist auch, daß bis heute über 25 Bestellungen eingegangen sind, 7 Leute allerdings einer Bezahlung völlig abgeneigt gegenüberstehen. Da dieses nichts Besonderes in der Zahlungsmoral ist - die Erfahrungen haben es mehrfach gezeigt - werden die LöBus nur gegen Vorkasse versandt. Ich bitte um Verständnis !

Leserbriefe:

"...daß die HMS-Funktion beim 41 nicht auf 8 oder 9 Stellen nach dem Komma genau arbeiten, war mir bekannt. Daß sich der Rundungsfehler jedoch auch bei dem für Zeitrechnungen eigentlich üblichen Anzeigeformat FIX 4 oder FIX 2 bemerkbar macht, hat mich doch etwas überascht.
Schalte den Rechner auf FIX 2 und führe folgende Operationen aus: 11,o5 ENTER 7,o5 HMS- . Das Ergebnis 3,6o steht deswegen in der Anzeige, weil der Rechner den Wert 3,595999999 auf 3,6o rundet. Das gleiche passiert z.B. bei 12,2o ENTER 8,1o HMS- im FIX 4 !
Umgehen läßt sich das Problem auch dann nicht, wenn man im Umweg über HR die Zahlen vorher umwandelt und mit "normalen" Rechenoperationen arbeitet. Sobald das Ergebnis mit HMS zurückgewandelt wird, taucht die unerwünschte Anzeige wieder auf.
Nobody is perfect ! Viktor Hinterleitner (448)

Möchte Ortsgruppe in Dortmund gründen: Macht alle mit ! Meldet Euch bei Mitgl. o93: Karsten Beierlein, Breierspfad 83, 46oo Dortmund-Wambel, Tel. o231/513o27

Fehlerbeschreibung - HP-41cv - Nr. 2133S11715

Nach Memory-Lost und SIZE o15 begann ich, ein Programm mit etwa 16oo Bytes einzugeben. Ungefähr 12oo Byte waren belegt, als der Rechner PACKING und TRY AGAIN ausführte. Danach startete ich die bereits vollständigen Unterprogramme. Der Rechner stoppte mit NONEXISTENT (Programmodus: XEQ 31). LBL 31 war verschwunden, an dessen Stelle und in den folgenden 53 Programmzeilen stand nun LBL o7 ???!!! Diese Zeilen ließen sich nicht löschen. Ich schrieb das Programm auf Karten und führte wieder MEMORY-LOST und SIZE 000 aus.
Seitdem steht danach im Programmodus nicht mehr 00 REG 319, sondern nur 00 REG 198. Peter Kark (467)

Anmerkung hierzu: Peter schrieb weiterhin, auch tagelanges Herausnehmen der Batterien führt zu keinem Erfolg. Wenn dem wirklich so ist, glaube ich fast, daß es sich hier um einen hardwaremäßigen Fehler handelt. Denn schließlich geht bei MEMORY-LOST, zumindest aber bei tagelangem Herausnehmen der Batterien, der gesamte RAM-Inhalt verloren. Selbst wenn Peter unbemerkt ein END mit o.g. Wirkung erzeugt haben sollte, es müßte dann doch auch verlorengehen. Oder ? Hat jemand ähnliche Erfahrungen gemacht ?
Oliver (1)

... möchte ich noch ein Prgr. einreichen, das zuerst die CAT3-Übersicht und anschließend die HEXCODE-Tabelle auflistet bis ..!??
LBL"AB" RCL e STO d END
Beim ersten mal wird gestartet mit 732o1 ASTO X STO e XEQ"AB" und, wenn der Cursor steht, immer R/S drücken. Das Programm kann mit dem Einschalten des PRGM-Modus beim Stillstand des Cursors angehalten werden und die in der Anzeige stehenden Befehle mit dem Drucker im TRACE-Modus aufgelistet. Burghard Schatz (3o5)

CLUBBÖRSE CLUBBÖRSE (dreimal in dieser Ausgabe !)

Suche PPC-ROM ! Ulrich Schwaderlap (438) Tel. o23o4/4o943
Suche PPC-ROM ! Otfried Werner (314) Tel. o611/728322
Suche PPC-ROM ! Christian Paschek (385) Tel. o7121/3786o
Suche PPC-ROM ! Miloslav Folprecht (345) Tel. o621/892494
Suche defekten HP-41c !! Miloslav Folprecht (345) Tel. o621/892494
Suche Mitglieder, die bereit sind, versch. Graphicen auf dem HP-Drucker zu erstellen. Barvergütung. Andreas Roemer, Eckenheimer Landstr. 3, 6ooo Frankfurt / M.
WER schreibt mir ein TI-59 Programm für den 41c um ?? Programmlisting und Genehmigung des Autors vorhanden. Siegfried Ludwig, 469o Herne 1, Stammstraße 41, Tel. o2323/53662
Gibt es HP-41/67/97 Anwender aus dem graphischen Gewerbe oder aus dem Verlagsgewerbe, die an Meinungs- und Informationsaustausch interessiert sind ? Bitte schreibt mir ! Siegfried Ludwig, Stammstr. 41, 469o Herne 1
Suche PPC-ROM ! Alberto Lenz (551), Nordstr. 23, CH-8oo6 Zürich

Wir suchen nebenberuflichen Seminarleiter für den HP-41 im Raum München ! Corvallis Team GmbH, Ortwin Rave, Postfach 1125, 6382 Friedrichsdorf 1, Tel. o6172/79551

Wir suchen zu sofort [illegible] Systemingenieur für HP-41 Produkt- und Softwarebetreuung hauptberuflich ! Corvallis Team GmbH, Ortwin Rave, Postfach 1125, 6382 Friedrichsdorf, Tel. o6172/79551

Verkaufe HP-41c mit drei Memorys (SDM) & Handbuch für 580,--DM.
Manfred Mickoteit (241), Am alten Hospital 9, 54oo Koblenz, Tel. o261/38469

B Ü C H E R E C K E

Titelverzeichnis der in der Clubbibliothek erhältlichen Bücher:

W.C. Wickes	"Synthetic Programming on the HP-41c"
H. Schumny	"Taschenrechner und Microcomputerjahrbuch 1981"
Gloistehn	"Programmieren von Taschenrechnern"
???	"EDV für Schulen, Bd. 1+2"
R. Eckert	"Die Programmierbaren von HP" (mit 2. Auflage über 41)
M. Gardner	"Die Zahlenspiele des Dr. Matrix"
M.Gardner	" Das Verhexte Alphabet"
"	"Rätsel und Denkspiele"
"	"Mathematik und Magie"
"	"Mathematische Rätsel und Spiele"
"	"Noch mehr mathematische Rätsel und Spiele"

Bild der Wissenschaft: Mathematisches Kabinett, Jahrgänge 1972-198o

HP-Programmsammlungen und Lösungsbücher:
Statistik, Geometrie, Spiele, Kalender, Maschinenbau, Chemie, Games-Pac 1 für HP-67/97, Games Pac 1 für HP-41, Network-Analyse-Pac (nur zur Einsichtnahme), Games-Pac (nur zur Einsichtnahme)

für TI: Lab-Chemistry-Lösungsbuch für TI-58/59

von unserem Mitglied Karlheinz Kraus (516) :

K. Kraus	"Der Einsatz programmierbarer Taschenrechner in der Arbeitsvorbereitung "
"	"Erstellen von Planzeiten über Regressionsrechnungen mittels programmierbarer Taschenrechner, Teil 1 bis 4"
"	"Maschinelle Kenngrößenberechnung von Arbeitsstudien und deren Genauigkeitsbeurteilung, Teil 1 bis 4"
"	"Programmpaket HP-67/97 für Arbeitsvorbereitung AV2"

Bei den Ausleihbedingungen tritt eine wichtige Neuerung ab sofort in Kraft: Werden die gewünschten Bücher nicht persönlich abgeholt, sondern per Post angefordert, so brauch in Zukunft keine Versandtasche mehr der Anforderung beigefügt werden. Die Kosten für 1oo Versandtaschen wurden vom Club übernommen, Bei der Rücksendung der Bücher ist aber unbedingt darauf zu achten, daß ein möglichst kleiner Adressaufkleber verwendet wird, so daß die Versandtasche durch überkleben eines anderen Adressaufklebers mehrmals benutzt werden kann.

Nach wie vor ist einer Anforderung auf dem Postwege unbedingt beizufügen : 1,6o DM in Briefmarken, Name und Adresse, Mitgliedsnummer. Fehlt eine dieser drei Sachen, so kann ein Ausleihwunsch leider nicht berücksichtigt werden !!!

Weiterhin können in Zukunft keine Vorbestellungen berücksichtigt werden, da die Bücher in der Reihenfolge des Eintreffens der Anforderungen verschickt werden.

Unser besonderer Dank gilt all jenen, die Bücher oder Photokopien gespendet und mit Rat und Tat zur Seite gestanden haben. Auch möchten wir uns hiermit bei all jenen bedanken und ihnen hiermit ein verschärftes Lob aussprechen, die bis jetzt Bücher ausgeliehen haben ! Die Ausleiher sind überdurchschnittlich vorsichtig mit den Büchern umgegangen, so daß es bis jetzt keine Ausfälle gegeben hat.

Wer spendet noch ein Buch ?? Alle guten Wünsche Frank (117) und Oliver (1)

ACHTUNG !!!

Ab dem 19. Oktober 1981 bin ich bis auf weiteres telefonisch nicht mehr zu erreichen. Die Clubanschrift bleibt aber nach wie vor Postfach 373, 2420 Eutin. Ich bitte daher, ab 19.1o.81 alle Korrespondenz mit mir schriftlich zu erledigen !

Oliver (1)

Ausblick 82

Als vor einem Jahr der Artikel "Ausblick 81" erschien, standen grundlegende Änderungen bevor. Ältere Mitglieder werden sich noch daran erinnern ... und feststellen, daß unsere damaligen Erwartungen eher noch übertroffen wurden. Doch darum soll es in diesem Bericht nicht gehen, sondern vielmehr um das, was im nächsten Jahr kommt. Die nachfolgenden Ankündigungen sind jedoch noch nicht endgültig entschieden, so daß ich auf die Unverbindlichkeit der genannten Überlegungen verweisen muß.

Durch ein freundliches Angebot von Rolf Hansmann (5oo) wird für 1982 folgendes vorgesehen: Die Clubinformationen werden auch 1982 beibehalten; das Thema beschränkt sich weiterhin auf den HP-41. Trotz aller Preiserhöhungen wird der Preis je Ausgabe 5,--DM betragen, wie 1981 auch schon. Nur erscheinen 1982 nicht acht Ausgaben, wie dieses Jahr, sondern ZWÖLF. Hierbei darf natürlich nicht übersehen werden, daß der Clubbeitrag keineswegs ausschließlich für die Infos entrichtet wird. Alle sonstigen Clubleistungen sind darin bereits voll enthalten. Ob die Infos als geleimte Lose-Blatt-Sammlung oder als geheftete Zeitschrift erscheinen steht noch nicht fest. Wenn machbar, soll die erste Möglichkeit gewählt werden.
Völlig sicher ist dagegen, daß die Infos nicht als "Hllig-Drucke" wie bisher erscheinen. Vielmehr werden sie gesetzt (d.h. ähnlich Tageszeitungen oder Bücher !), was zwar deutlich höhere Kosten mit sich bringt, dafür aber auch zwei bis viermal soviel Informationen auf gleichem Raum unterbringt. Das Format wird DIN A4 bleiben. Die mit der Fotosatzmaschine von Rolf (5oo) erstellten Infos werden dann Offsett gedruckt, was zu einer wesentlich besseren Druckqualität (wie Bücher !) führt.
Die so zwölfmal jährlich erscheinenden Infos werden per Postvertriebsstück oder als Streifbandzeitung verschickt. Damit brauch uns die Gebührenerhöhung für Drucksachen/Büchersendungen nicht mehr zu interessieren.
Allein durch die Portoersparnis finanzieren wir dann mind. eine ganze Ausgabe ! Die Frage, wieviel Seiten die Infos dann wohl umfassen werden, vermag ich aber nicht zu sagen. Das hängt nämlich weitgehend mit der Aktivität der Mitglieder zusammen.
Die Beiträge sollen Anfang Januar erstattet werden, damit bereits zum Jahresanfang große Papiermengen eingekauft werden können, um einen günstigeren Seitenpreis zu erhalten. Außerdem ist die Zahlungsmoral der Mitglieder auch nicht sonderlich berauschend ....

Der Beitrag von 6o,--DM p.a. gilt für alle inländischen Mitglieder. Der Zuschlag für Ausländer hängt von den Portomehrkosten ab, die wir noch erfragen müssen. Der Betrag sollte zwischen dem 1. und 16. Januar 1982 bezahlt werden, da unter allen in dieser Zeit eingehenden Überweisungen als besonderer Anreiz zum pünktlichen Bezahlen zwei Module für den 41 noch eigener Wahl verlost werden. Ein Überweisungsvordruck für Postscheck- und Bankkonten, die auch zur Bareinzahlung verwendbar sind, kommt mit dem Dezemberinfo. Wer einen Gutschein besitzt, kann diesen mit dem Beitrag verrechnen und zusammen mit einem Scheck über den Differenzbetrag einsenden. Ein besonderer Service gilt für Studenten etc., die meinen, ihr Bankkonto hält solche Belastung zum Jahresanfang nicht aus: Der Beitrag kann auch per V-Scheck entrichtet werden an Oliver (1). Auf dem Scheck ist deutlich die Mitgliedsnummer zu vermerken. Der Vorteil liegt darin, daß diese Schecks dann erst zum 1.3.82 eingelöst werden, trotzdem aber an o.g. Verlosung teilnehmen ! Aber bitte: Es sende nur einen Scheck, wer das Geld nicht Anfang Januar auf dem Konto hat ! Bitte jetzt keinesfalls im voraus bezahlen ! Unbedingt die Vordrucke im Dezemberinfo abwarten !

Alle anderen Clubleistungen, ob nun Clubbibliothek, Magnetkartenbibliothek oder Clubtreffen, bleiben bestehen und werden weiter ausgebaut. Neuen Unterfangen steht nichts im Wege.
Auch soll der Club für kommerzielle HP-41 Nutzer interessanter werden: Die letzte Seite der Clubinfos soll Firmen für Werbezwecke zur Verfügung stehen. Aber wie gesagt: Nur die letzte Seite ! Für das Januarinfo ist ferner die Vorstellung aller professionellen Software für den HP-41, die in der Bundesrepublik erhältlich ist, geplant: Marktübersicht Software ! Auch sollen Testberichte und Neuvorstellungen der kommenden Peripherieeinheiten größeren Raum als bisher einnehmen. Ab Januar 82 sollen ferner alle Programme, die in prisma erscheinen, auch in Strichcodes abgedruckt werden. Dies wird durch die freundlicherweise von HP Frankfurt zur Verfügung gestellte Anlage ermöglicht, die Strichcodes direkt aus den Magnetkarten des HP-41 erstellt. Und .. Und ... und &&&&& euer oliver (1)

Liebe Clubmitglieder!

Wie Ihr vielleicht im letzten Info gelesen habt, werde ich mich ab jetzt in unserem Club als "Projektleiter Mathematik" betätigen. Um mir diese Arbeit etwas zu erleichtern, möchte ich Euch bitten, nach Möglichkeit folgende Punkte zu beachten:

1. Bitte legt allen Programmen ein gedrucktes oder getipptes Listing bei.
2. Wer seine Programme, falls sie nicht veröffentlicht werden, wiederhaben will, der lege bitte Rückporto bei.
3. Wer mir einen ganz großen Gefallen tun will, füge beschriebene Magnetkarten bei(In diesem Fall das Rückporto nicht vergessen!!).

Ansonsten freue ich mich auf eure Post. Falls jemand Interesse an einem Mathe-Lösungsbuch hat, sollte er mir mal schreiben, welche Programme er gern darin vorfinden und was dafür anzulegen(grobe Preisvorstellung) er bereit wäre.

Nach diesem Gelabere möchte ich nun endlich in medias res gehen und Euch die nachfolgenden Programme etwas näher bringen:

1) VEKTOR werden wohl hauptsächlich die streßgeplagten Schüler benutzen, um sich folgende Berechnungen durchführen zu lassen:
   1) Einheitsvektor VØ
   2) Betrag eines Vektors /V/
   3) a) Kreuzprodukt V1XV2
      b) Spatprodukt SPAT
      c) Skalarprodukt V1xV2
   4) Vektoraddition V1+V2
   5) Vektorsubtraktion V1-V2
   6) Volumen eines Vierflachs 4FLACH
   7) Winkel zwischen zwei Vektoren V1 V2
   8) Koordinatentransformation
      a) Polar-Kartesisch P-C
      b) Kartesisch-Polar C-P

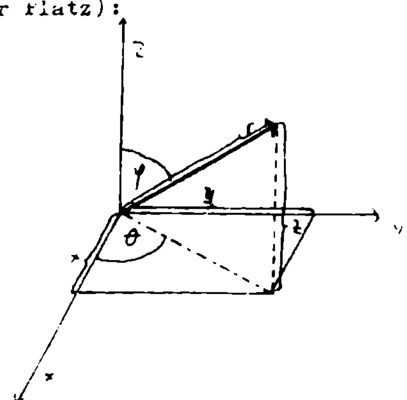
2) Noch'n PPZ-Programm

Für Leute, die es eilig haben, denen aber F. Hillebrandts Programm zu lang war. Sehr zu begrüßen ist das beiliegende Programm "T", das eine recht vernünftige Schätzung der Höchstrechenzeit liefert(Die Zahl in Zeile 10 kann je nach Modell etwas variieren).

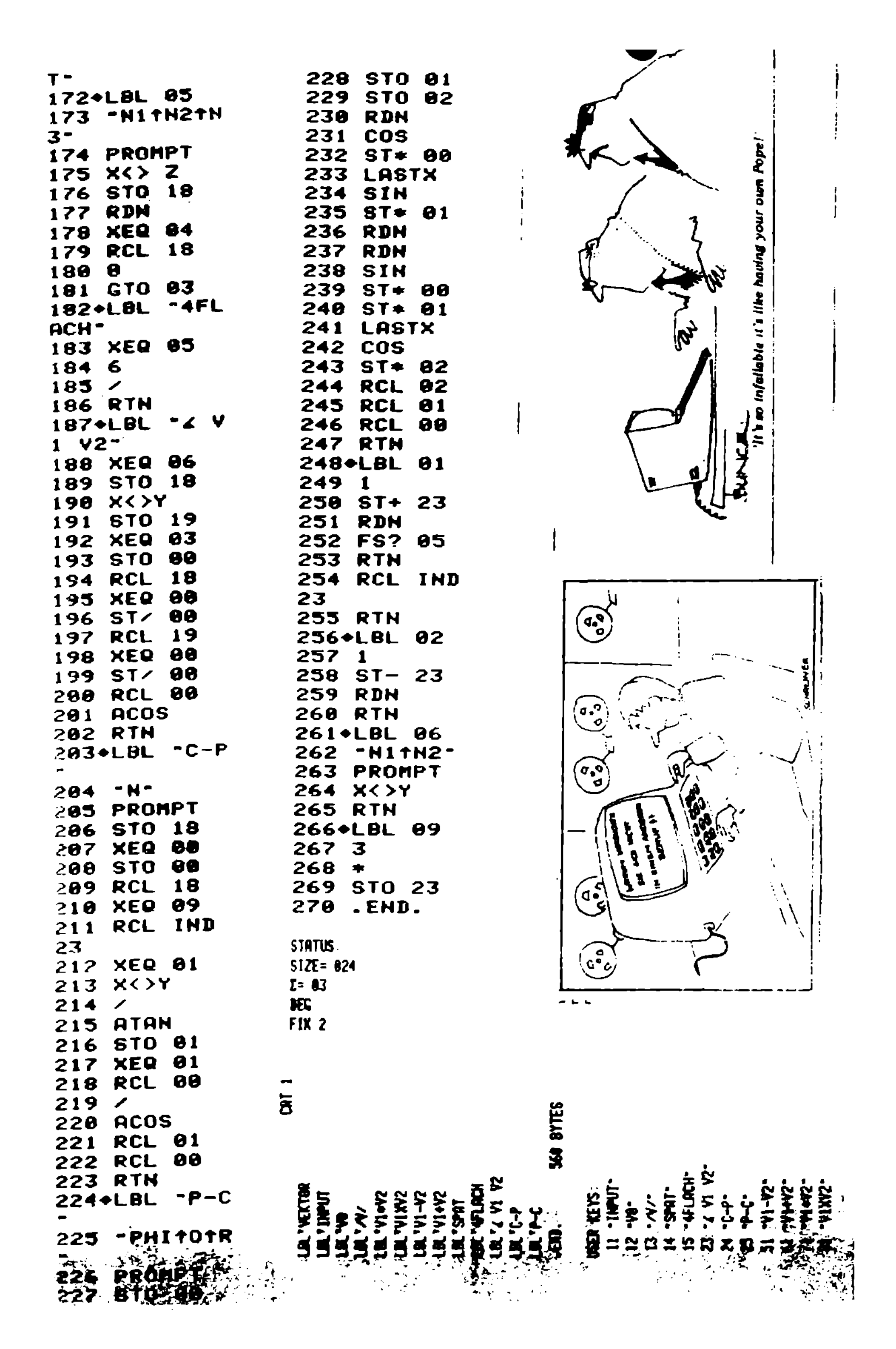
3) Gleichungen 4. Grades

Kann man zwar auch mit dem Mathe-Modul Programm "POLY" berechnen, aber dabei wird eine Iteration verwandt, deren Rechenzeiten einen mitunter nicht gerade vom Hocker hauen, wohingegen das folgende Programm die Lösung direkt berechnet.

Nun noch eine Skizze zum VEKTOR-Programm(für weitere Formeln o.ä. fehlt mir der Platz):



```
 01♦LBL "VEK
TOR"
 02♦LBL "INP
UT"
 03♦LBL 07
 04 "N↑X↑Y↑Z
"
 05 PROMPT
 06 X<> T
 07 X<>Y
 08 X<> Z
 09 X<>Y
 10♦LBL a
 11 SF 05
 12 STO 23
 13 RDN
 14 X<> 23
 15 3
 16 *
 17 X<> 23
 18 STO IND
23
 19 XEQ 01
 20 RDN
 21 STO IND
23
 22 XEQ 01
 23 RDN
 24 STO IND
23
 25 CF 05
 26 STOP
 27 GTO 07
 28♦LBL "V0"
 29 "N"
 30 PROMPT
 31 STO 22
 32 XEQ 00
 33 1/X
 34 STO 21
 35 RCL 22
 36 XEQ 09
 37 2
 38 +
 39 STO 23
 40 RCL IND
23
 41 RCL 21
 42 *
 43 XEQ 02
 44 RCL IND
23
 45 RCL 21
 46 *
 47 XEQ 02
 48 RCL IND
23
 49 RCL 21
 50 *
 51 RTN
 52♦LBL "[illegible]
"
 53 "N"
 54 PROMPT
 55♦LBL 00
 56 XEQ 09
 57 RCL IND
23
 58 XEQ 01
 59 R-P
 60 XEQ 01
 61 R-P
 62 RTN
 63♦LBL "V1*
V2"
 64 XEQ 06
 65♦LBL 03
 66 XEQ 09
 67 STO 22
 68 RDN
 69 XEQ 09
 70 RCL IND
23
 71 XEQ 01
 72 XEQ 01
 73 RCL 22
 74 STO 23
 75 RDN
 76 RCL IND
23
 77 R↑
 78 *
 79 XEQ 01
 80 R↑
 81 *
 82 +
 83 XEQ 01
 84 R↑
 85 *
 86 +
 87 RTN
 88♦LBL "V1X
V2"
 89 XEQ 06
 90♦LBL 04
 91 STO 22
 92 RDN
 93 XEQ 09
 94 RCL IND
23
 95 STO 19
 96 XEQ 01
 97 STO 20
 98 XEQ 01
 99 STO 21
100 RCL 22
101 XEQ 09
102 RCL IND
23
103 STO 02
104 XEQ 01
105 STO 00
106 XEQ 01
107 STO 01
108 RCL 19
109 [illegible]
110 RCL 00
111 RCL 21
112 *
113 -
114 RCL 01
115 X<>Y
116 STO 01
117 RDN
118 RCL 20
119 *
120 RCL 00
121 RCL 21
122 *
123 -
124 CHS
125 RCL 00
126 X<>Y
127 STO 00
128 RDN
129 RCL 19
130 *
131 RCL 02
132 RCL 20
133 *
134 -
135 CHS
136 STO 02
137 RCL 01
138 RCL 00
139 RTN
140♦LBL "V1-
V2"
141 SF 00
142♦LBL "V1+
V2"
143 XEQ 06
144 STO 22
145 RDN
146 XEQ 09
147 RCL IND
23
148 FS? 00
149 CHS
150 STO 00
151 XEQ 01
152 FS? 00
153 CHS
154 STO 01
155 XEQ 01
156 FS?C 00
157 CHS
158 STO 02
159 RCL 22
160 XEQ 09
161 RCL IND
23
162 ST+ 00
163 XEQ 01
164 ST+ 01
165 XEQ 01
166 ST+ 02
167 RCL 02
168 RCL 01
169 RCL 00
170 RTN
171♦LBL "SPA[illegible]
```



```
T"
172+LBL 05
173 "N1↑N2↑N
3"
174 PROMPT
175 X<> Z
176 STO 18
177 RDN
178 XEQ 04
179 RCL 18
180 0
181 GTO 03
182+LBL "4FL
ACH"
183 XEQ 05
184 6
185 /
186 RTN
187+LBL "∡ V
1 V2"
188 XEQ 06
189 STO 18
190 X<>Y
191 STO 19
192 XEQ 03
193 STO 00
194 RCL 18
195 XEQ 00
196 ST/ 00
197 RCL 19
198 XEQ 00
199 ST/ 00
200 RCL 00
201 ACOS
202 RTN
203+LBL "C-P
"
204 "N"
205 PROMPT
206 STO 18
207 XEQ 00
208 STO 00
209 RCL 18
210 XEQ 09
211 RCL IND
23
212 XEQ 01
213 X<>Y
214 /
215 ATAN
216 STO 01
217 XEQ 01
218 RCL 00
219 /
220 ACOS
221 RCL 01
222 RCL 00
223 RTN
224+LBL "P-C
"
225 "PHI↑0↑R
"
226 PROMPT
227 STO 00
228 STO 01
229 STO 02
230 RDN
231 COS
232 ST* 00
233 LASTX
234 SIN
235 ST* 01
236 RDN
237 RDN
238 SIN
239 ST* 00
240 ST* 01
241 LASTX
242 COS
243 ST* 02
244 RCL 02
245 RCL 01
246 RCL 00
247 RTN
248+LBL 01
249 1
250 ST+ 23
251 RDN
252 FS? 05
253 RTN
254 RCL IND
23
255 RTN
256+LBL 02
257 1
258 ST- 23
259 RDN
260 RTN
261+LBL 06
262 "N1↑N2"
263 PROMPT
264 X<>Y
265 RTN
266+LBL 09
267 3
268 *
269 STO 23
270 .END.

STATUS
SIZE= 024
I= 03
DEG
FIX 2
```

Folgende B[illegible]spielaufgaben sind zu lösen:

1) Die Richtung einer Ebene ist gegeben durch die Vektoren

$$\mathfrak{a} = \begin{pmatrix} 5 \\ 3 \\ -1 \end{pmatrix} \quad \text{und} \quad \mathfrak{b} = \begin{pmatrix} -7 \\ 1 \\ 4 \end{pmatrix} .$$

   Berechnen Sie den Lotvektor der Ebene und überprüfen Sie das Ergebnis indem Sie
   a) das Skalarprodukt von jedem der Ausgangsvektoren mit dem Lotvektor bilden (=0),
   b) die Winkel zwischen dem Lotvektor und den beiden Ausgangsvektoren bestimmen.

2) Wie lang ist der Lotvektor aus 1)?

3) Wie lautet der Einheitsvektor des Lotvektors aus 1)?

4) Bestimmen Sie die Winkel φ und ϑ des Vektors

$$\mathfrak{v} = \begin{pmatrix} 3 \\ 1 \\ -2 \end{pmatrix}$$

   Überprüfen Sie das Ergebnis mit der Funktion ∢ V1 V2

5) Ein Vierflach ist gegeben durch die Punkte
   P_1 (1/2/0), P_2 (2/3/1), P_3 (1/5/3), P_4 (1/2/1)
   Berechnen Sie das Volumen.

6) Eine Ebene ist gegeben durch

$$\mathfrak{x}_1 = \begin{pmatrix} 6 \\ 3 \\ 7 \end{pmatrix} + \mu \begin{pmatrix} 2 \\ -2 \\ -4 \end{pmatrix} + \nu \begin{pmatrix} -1 \\ 3 \\ -16 \end{pmatrix}$$

   eine Gerade ist gegeben durch

$$\mathfrak{g} : \mathfrak{x}_2 = \begin{pmatrix} 3 \\ 7 \\ 9 \end{pmatrix} + \lambda \begin{pmatrix} -6 \\ 1 \\ -5 \end{pmatrix}$$

   Bestimmen Sie den Schnittpunkt.

Das Programm:

Das Programm ist so aufgebaut, daß jede Funktion einer Taste zugeordnet ist, (vgl. USER KEYS) und durch Drücken dieser Taste aufgerufen wird. Der oder die Vektoren müssen vorher über die Funktion INPUT eingegeben werden, es können bis zu fünf Vektoren gleichzeitig im Rechner gespeichert werden. Die Eingabe erfolgt in der Form N↑X↑Y↑Z, wobei N die Nummer des Vektors ist und X,Y,Z die entsprechneden Koordinaten sind.
Um Vektoren, die als Zwischenergebnisse abgespeichert werden sollen einzugegeben, gibt man die laufende Nummer des Vektors ein und drückt f A.
Führt man eine arithmetischen Funktion, (nicht INPUT) aus, fragt der Rechner nach der oder den Nummer der Vektoren und führt sie aus. Anschließend enthalten die Stackregister die entsprechenden Koordinaten, X in X, Y in Y, und Z in Z.

Lösungen der Beispielaufgaben:

(1.)

```
             XEQ "INPUT"
NYXYYZ
             1,00 ENTER↑
             5,00 ENTER↑
             3,00 ENTER↑
            -1,00    RUN
             XEQ "INPUT"
NYXYYZ
             2,00 ENTER↑
            -7,00 ENTER↑
             1,00 ENTER↑
             4,00    RUN
             XEQ "V1XV2"
N1↑N2
             1,00 ENTER↑
             2,00    RUN
                   PRSTK

T=   3,00
Z=  26,00
Y= -13,00
X=  13,00
```

→ (13, -13, 26)

```
             3,00  XEQ a
```

a)

```
             XEQ "V1*V2"
N1↑N2
             1,00 ENTER↑
             3,00    RUN
             0,00    ***

             XEQ "V1*V2"
N1↑N2
             2,00 ENTER↑
             3,00    RUN
             0,00    ***
```

b)

```
             XEQ "∠ V1 V2"
N1↑N2
             1,00 ENTER↑
             3,00    RUN
            90,00    ***

             XEQ "∠ V1 V2"
N1↑N2
             2,00 ENTER↑
             3,00    RUN
            90,00    ***
```

(2.)

```
             XEQ "/V/"
N
             3,00    RUN
            31,84    ***
```

(3.)

```
             XEQ "V0"
N
             3,00    RUN
                   PRSTK
```

(4.)

```
             XEQ "INPUT"
NYXYYZ
             1,00 ENTER↑
             2,00 ENTER↑
             1,00 ENTER↑
            -2,00    RUN
             XEQ "C"
N
             1,00    RUN
                   PRSTK

T=   18,43
Z=  122,31   φ
Y=   18,43   ϑ
X=    3,74   r

             XEQ "INPUT"
NYXYYZ
```

z-Achse =

```
             2,00 ENTER↑
             0,00 ENTER↑
             0,00 ENTER↑
             1,00    RUN
             XEQ "∠ V1 V2"
N1↑N2
             1,00 ENTER↑
             2,00    RUN
           122,31    ***

             XEQ "INPUT"
NYXYYZ
```

Projektion von a auf xy-Ebene =

```
             3,00 ENTER↑
             3,00 ENTER↑
             1,00 ENTER↑
             0,00    RUN
             XEQ "INPUT"
NYXYYZ
```

x-Achse =

```
             4,00 ENTER↑
             1,00 ENTER↑
             0,00 ENTER↑
             0,00    RUN
             XEQ "∠ V1 V2"
N1↑N2
             3,00 ENTER↑
             4,00    RUN
            18,43    ***
```

([illegible])

```
             XEQ "INPUT"
NYXYYZ
             1,00 ENTER↑
             1,00 ENTER↑
             2,00 ENTER↑
             1,00    RUN
             XEQ "INPUT"
NYXYYZ
             2,00 ENTER↑
             2,00 ENTER↑
             3,00 ENTER↑
             1,00    RUN
             XEQ "INPUT"
NYXYYZ
             3,00 ENTER↑
             1,00 ENTER↑
             5,00 ENTER↑
             3,00    RUN
             XEQ "INPUT"
NYXYYZ
             4,00 ENTER↑
             1,00 ENTER↑
             2,00 ENTER↑
             1,00    RUN
```

Berechnung d. Seitenvektoren:

```
             XEQ "V1-V2"
N1↑N2
             2,00 ENTER↑
             1,00    RUN
             2,00  XEQ a

             XEQ "V1-V2"
N1↑N2
             3,00 ENTER↑
             1,00    RUN
             3,00  XEQ a

             XEQ "V1-V2"
N1↑N2
             4,00 ENTER↑
             1,00    RUN
             4,00  XEQ a

             XEQ "4FLACH"
N1↑N2↑N3
             2,00 ENTER↑
             3,00 ENTER↑
             4,00    RUN
             0,50    ***
```

Lösung d[illegible]6. Aufgabe:

Durch Gleichsetzen entsteht folgendes Gleichungssystem:

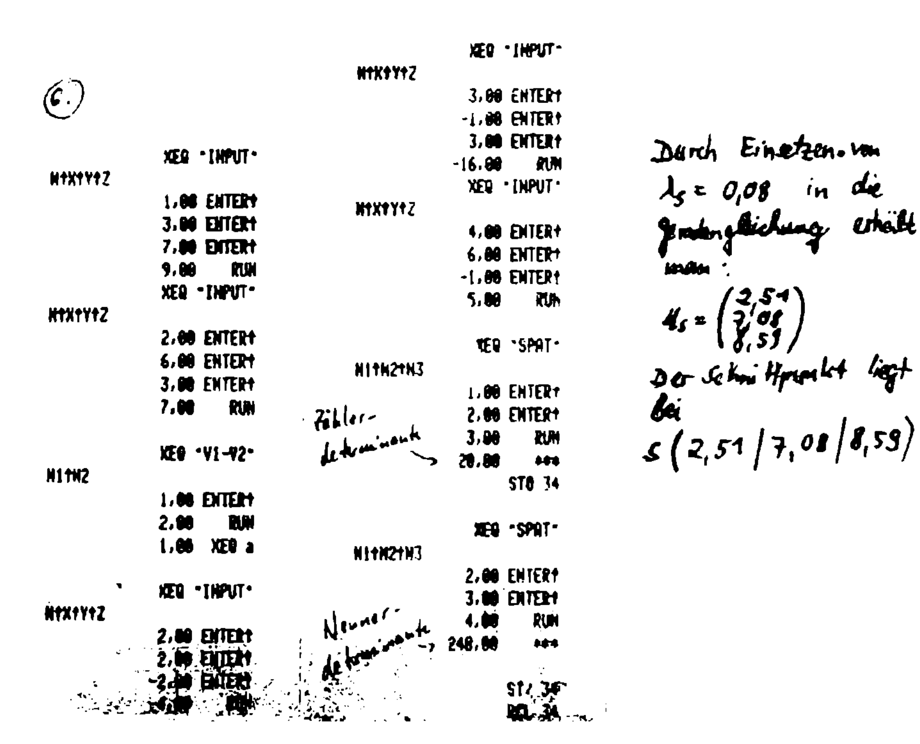
$$\begin{pmatrix}3\\7\\9\end{pmatrix} + \lambda_s\begin{pmatrix}-6\\1\\-5\end{pmatrix} = \begin{pmatrix}6\\3\\7\end{pmatrix} + \mu_s\begin{pmatrix}2\\-2\\-4\end{pmatrix} + \nu_s\begin{pmatrix}-1\\3\\-16\end{pmatrix}$$

Daraus folgt

$$\begin{pmatrix}3\\7\\9\end{pmatrix} - \begin{pmatrix}6\\3\\7\end{pmatrix} = \mu_s\begin{pmatrix}2\\-2\\-4\end{pmatrix} + \nu_s\begin{pmatrix}-1\\3\\-16\end{pmatrix} + \lambda_s\begin{pmatrix}6\\-1\\5\end{pmatrix}$$

$$\Rightarrow \quad \lambda_s = \frac{\left|\left[\begin{pmatrix}3\\7\\9\end{pmatrix}-\begin{pmatrix}6\\3\\7\end{pmatrix}\right] \begin{pmatrix}2\\-2\\-4\end{pmatrix} \begin{pmatrix}-1\\3\\-16\end{pmatrix}\right|}{\left|\begin{pmatrix}2\\-2\\-4\end{pmatrix} \begin{pmatrix}-1\\3\\-16\end{pmatrix} \begin{pmatrix}6\\-1\\5\end{pmatrix}\right|}$$

$$= \frac{\left[\begin{pmatrix}3\\7\\9\end{pmatrix}-\begin{pmatrix}6\\3\\7\end{pmatrix}\right] \cdot \left[\begin{pmatrix}2\\-2\\-4\end{pmatrix} \times \begin{pmatrix}-1\\3\\-16\end{pmatrix}\right]}{\begin{pmatrix}2\\-2\\-4\end{pmatrix} \cdot \left[\begin{pmatrix}-1\\3\\-16\end{pmatrix} \times \begin{pmatrix}6\\-1\\5\end{pmatrix}\right]}$$



G L E I C H U N G E N 4. G R A D E S

Das Programm löst algebraisch die Gleichung

$$ax^4+bx^3+cx^2+dx+e=0 \qquad \text{mit } a,b,c,d,e \in R$$

$$x \in R \text{ und } a \neq 0$$

Das Programm berechnet alle Lösungen (vier) incl. komplexe der Gleichung. Es sind weiterhin Gleichungen 2. und 3. Grades mit dem Programm zu berechnen. Die entsprechenden Gleichungen müssen dann mit x^2 bzw. x multipliziert werden. Die Lösungen (0;0) bzw. (0) sind nicht zu berücksichtigen.
Es werden neun Speicher benötigt (SIZE 009).

Schritt	Instruktionen	Eingabe	Anzeige
1	Start	XEQ"ROOT"	a=?
2	Eingabe der Koeffizienten	a R/S	b=?
3	"	b R/S	c=?
4	"	c R/S	d=?
5	"	d R/S	e=?
6	"	e R/S	
7	Anzeige der 1. Lösung		X1=...
8	Anzeige der 2. Lösung	R/S	X2=...
9	Anzeige der 3. Lösung	R/S	X3=...
10	Anzeige der 4. Lösung	R/S	X4=...
11	Weiter bei 7	R/S	X1=...
[illegible]	FIX 9	CLX	Realteil
		X Y	Imaginärteil

Beispiele

$x^4+x^3-\frac{25}{4}x^2-\frac{1}{4}x+1,5=0$ mit a=1, b=1, c=-6,25, d=-0,25, e=1,5 — Lsg: X1=-3, X2=-0,5, X3=0,5, X4=2

$x^3-8x^2+19x-12=0$ mit a=1, b=-8, c=19, d=-12, e=0 — Lsg: X1=0 nicht zu berücks., X2=1, X3=3, X4=4

```
01 LBL"ROOT"
02 "a=?"
03 PROMPT
04 STO 03 <1/X
05 STO 02
06 STO 01
07 STO 00
08 "b=?"
09 PROMPT
10 STx 00
11 "c=?"
12 PROMPT
13 STx 01
14 "d=?"
15 PROMPT
16 STx 02
17 "e=?"
18 PROMPT
19 STx 03
20 RCL 02
21 STO 07
22 X↑2
23 RCL 00
24 STx 07
25 X↑2
26 RCL 03
27 x
28 +
29 RCL 01
30 CHS
31 STO 06
32 RCL 03
33 4
34 x
35 ST- 07
36 x
37 +
38 CHS
39 STO 08
40 RCL 06
41 3
42 /
43 STO 06
44 RCL 07
45 x
46 RCL 08
47 -
48 2
49 /
50 RCL 06
51 3
52 Y↑X
53 -
54 STO 05
55 X↑2
56 RCL 07
57 3
58 /
59 RCL 06
60 X↑2
61 -
62 3
63 Y↑X
64 +
65 X>0?
66 GTO 00
67 CHS
68 SQRT
69 RCL 05
70 R-P
71 3
72 1/X
73 Y↑X
74 X<>Y
75 3
76 /
77 X<>Y
78 P-R
79 X<>Y
80 3
81 SQRT
82 x
83 2
84 /
85 STO 07
86 CHS
87 STO 08
88 X<>Y
89 ST- 08
90 ST- 07
91 2
92 x
93 RCL 08
94 X>Y?
95 X<>Y
96 RDN
97 RCL 07
98 X<=Y?
99 X<>Y
100 GTO 02
101 LBL 00
102 SQRT
103 ST+ 05
104 CHS
105 2
106 x
107 RCL 05
108 +
109 ENTER↑
110 SIGN
111 X<>Y
112 ABS
113 3
114 1/X
115 Y↑X
116 x
117 RCL 05
118 SIGN
119 RCL 05
120 ABS
121 3
122 1/X
123 Y↑X
124 x
125 +
126 LBL 02
127 RCL 06
128 -
129 STO 07
130 2
131 /
132 STO 08
133 ENTER↑
134 X↑2
135 RCL 03
136 -
137 ENTER↑
138 ABS
139 +
140 2
141 /
142 SQRT
143 ST-06
144 +
145 STO 06
146 RCL 00
147 2
148 /
149 STO 05
150 RCL 05
```

```
151 X↑2
152 RCL 07
153 +
154 RCL 01
155 -
156 SQRT
157 ST- 05
158 +
159 STO 07
160 RCL 06
161 x
162 RCL 05
163 RCL 08
164 x
165 +
166 RCL 02
167 -
168 ABS
169 RCL 07
170 RCL 08
171 x
172 RCL 05
173 RCL 06
174 x
175 +
176 RCL 02
177 -
178 ABS
179 X>Y?
180 GTO 03
181 RCL 08
182 X<> 06
183 STO 08
184 LBL 03
185 CF 05
186 "X1"
187 7
188 XEQ 04
189 SF 05
190 "X2"
191 7
192 XEQ 04
193 CF 05
194 "X3"
195 5
196 XEQ 04
197 SF 05
198 "X4"
199 5
200 XEQ 04
201 GTO 03
202 LBL 04
203 STO 04
204 RCL IND 04
205 2
206 /
207 X↑2
208 ISG 04
209 X<> Y
210 RCL IND 04
211 DSE 04
212 X<> Y
213 -
214 0
215 X<>Y
216 X>0?
217 GTO 05
218 ABS
219 SQRT
220 X<>Y
221 LBL 05
222 SQRT
223 FS? 05
224 GTO 06
225 CHS
226 X<>Y
227 CHS
228 X<>Y
229 LBL 06
230 RCL IND 04
231 2
232 /
233 -
234 FIX 2
235 "⊢="
236 ARCL X
237 X<>Y
238 X=0?
239 GTO 01
240 X>0?
241 "⊢+"
242 ARCL X
243 "⊢I"
244 LBL 01
245 X<>Y
246 FIX 9
247 PROMPT
248 RTN
```

Ein Primzahlprogramm -- mal wieder !

Zuerst waren es die Size-Routinen, vor denen sich weder Key-Notes noch prisma retten konnte. Jetzt sind es die Primzahlprogramme. In der Methode sind sie überwiegend gleich: Eine große Schleife, in der Teiler berechnet werden, die ihrerseits nicht durch 2,3 oder 5 teilbar sind. Lediglich das Programm von F.Hillebrandt verwendet eine andere Methode, und es ist das schnellste und auch längste der veröffentlichten Programme.
Der Grund für das große Interesse an solchen Routinen dürfte der sein: ebenso wie bei Size läßt sich zwar immer eine Information aus dem Programmlauf herausziehen, aber der Zeitaufwand läßt sich nicht vom Anwender bestimmen, weil er nur von der eingegebenen Zahl abhängt.
Das folgende Programm ist nach Vorschlägen von F. Hillebrandt und eigenen Ideen aus dem Primzahlenprogramm von U. Davertzhofen und mir hervorgegangen. Im Beispiel 1 000 003 benötigt dieses Programm mit ca. 40 sec. (vorm. 1 min. 10 sec.) etwa 42 % weniger Zeit, und geht auch mit den Bytes sparsamer um. Wird eine 0 eingegeben, folgt keine Endlosschleife, sondern ein DATA ERROR, ebenso bei negativen oder gebrochenen Eingaben. SIZE 006, 200 Bytes.
Noch ein Hinweis: Die Schleife reizt zur Verwendung eines Unterprogramms, was aber für den Geschwindigkeitsvorteil absolut tödlich ist, und außerdem kaum mehr als 2 oder 3 Register beschert.
Die Eingabe einer Zahl erfolgt über XEQ"PRF" oder XEQ"A", die Ausgabe der Teiler mit R/S, bis die Meldung ENDE erscheint.
Das Programm T gibt einen ziemlich genauen Hinweis auf die Laufzeit: Für Zahlen um 10^10 weniger als 1 Std. 5 min.

```
LISTING DES ZWEIT-
SCHNELLSTEN PRIMZAHLEN-
SUCHERS NACH DEM VON
F. HILLEBRANDT
                PRP "PRF"

01*LBL "PRF"
02*LBL A
CLRG  LN  LASTX  STO 00
FRC  FACT  2  STO 01  4
STO 02  +  STO 03
RCL 00  ENTER↑  ENTER↑
SF 00  RCL 01  MOD  X=0?
XEQ IND 01  CLX  3  MOD
X=0?  XEQ IND 01  CLX  5
MOD  X=0?  XEQ IND 01
SIGN  SIGN

35*LBL 00
X<> L  RCL 03  +  MOD
X=0?  XEQ IND 01  X<> L
RCL 02  +  MOD  X=0?
XEQ IND 01  X<> L
RCL 01  +  MOD  X=0?
XEQ IND 01  X<> L
RCL 02  +  MOD  X=0?
XEQ IND 01  X<> L
RCL 01  +  MOD  X=0?
XEQ IND 01  X<> L
RCL 02  +  MOD  X=0?
XEQ IND 01  X<> L
RCL 03  +  MOD  X=0?
XEQ IND 01  X<> L
RCL 01  +  MOD  X=0?
XEQ IND 01  X<> L  X↑2
X<=Y?  GTO 00  1  RCL 00
FC?C 00  GTO 01  X≠Y?
XEQ IND 02  "ENDE"
PROMPT

96*LBL 01
"PRIM"  PROMPT

99*LBL 02
FS? 00  GTO 03  RDN
LASTX  X≠Y?  RTN  SF 00

107*LBL 03
1  ST+ 05  RCL 00  LASTX
/  STO 00  LASTX  MOD
X=0?  GTO 03  LASTX

119*LBL 04
STO 04  FIX 0  CF 29
"  "  ARCL 04  CLX
X<> 05  1  X<Y?  "⊢ ↑"
X<Y?  ARCL Y  FIX 2
SF 29  RCL 04  PROMPT
RCL 04  SIGN  RCL 00
ENTER↑  ENTER↑  ENTER↑
END

                PRP "T"

01*LBL "T"
02*LBL A
FIX 4  SQRT  30  /  INT
1  +  3886       HMS
.END.
```

Ralf Pfeifer
Rube... 5
5000 Köln - 50
Tel. 35 20 34

ACHTUNG: Die Zahlen, die "T" berechnet, werden in der gleichen Form wie HMS ausgegeben!

286,287-81

S T A R   T R E K

Dieses Programm basiert auf dem Programm MONDLANDUNG von Oliver Rietschel.

Das Programm simuliert die Landung des Raumschiffes ENTERPRISE auf einem beliebigen Planeten. Der Spieler wird aufgefordert, die Kontrolle über das Raumschiff zu übernehmen. Seine Aufgabe ist es, das Schiff so weich wie möglich auf dem Planeten zu landen, sekundär dabei so wenig Treibstoff wie möglich zu verbrauchen. Er befindet sich in einer Höhe von 500 Metern, die Fallgeschwindigkeit beträgt -50 m/s, der Treibstoffvorrat besteht aus 50 Einheiten. Durch die Eingabe von Treibstoff wird ein Bremsschub ausgelöst, der die Fallgeschwindigkeit senkt, bzw. die Steiggeschwindigkeit erhöht. Die Eingabe von Treibstoffeinheiten ist nur möglich, wenn der Count Down Null erreicht hat. Für die Eingabe bleibt eine Sekunde Zeit bis zur Zündung. Man ist allerdings nicht gezwungen eine Eingabe auszuführen. Sollte dabei aber die Geschwindigkeit zu gross werden, verglüht das Raumschiff. Der Pilot wird rechtzeitig durch die Anzeige "** DANGER **" darauf hingewiesen. Das gleiche gilt, wenn der Treibstoffvorrat in einen kritischen Bereich sinkt.

Das Spiel ist weiterhin so aufgebaut, dass der Rechner bei jeder erfolgreichen Landung den nach der Landung verbliebenen Treibstoffrest mit dem besten Wert aus den vorhergehenden Spielen vergleicht und gegebenenfalls die gerade erfolgte Landung als die bisher Beste ausweist. Der Rechner zeigt ebenfalls die Anzahl der bisher erfolgten Spiele an.

Das Display hat folgendes Format:

XXX YY ZZ T

- XXX = Flughöhe
- YY = Fallgeschwindigkeit
- ZZ = Treibstoffvorrat
- T = Count Down: 3 - 2 - 1 - 0
  Spielende : E

Meldungen bei einer Landung:

- TOUCH DOWN : weiche Landung
- CRASH DOWN : Bruchlandung
- BURNT DOWN : Verglühen des Raumschiffes
- PILOT DOWN : Eingabefehler des Piloten

```
01♦LBL "AE"
02 FIX 0
03 CF 29
04 500
05 STO 00
06 -50
07 STO 01
08 50
09 STO 02
10 " "
11 ASTO 04
12 " ENTER"
13 ASTO 05
14 "PRISE"
15 ASTO 06
16 " STAR TREK"
17 PROMPT
18 " TAKE OVER"
19 AVIEW
20 PSE
21 XEQ 09
22 " CONTROLL"
23 AVIEW
24 PSE
25♦LBL 01
26 RCL 01
27 ABS
28 56
29 X<=Y?
30 XEQ 07
31 RDN
32 61
33 X<=Y?
34 GTO 04
35 XEQ 10
36 "├3"
37 AVIEW
38 PSE
39 XEQ 10
40 "├2"
41 AVIEW
42 PSE
43 XEQ 10
44 "├1"
45 AVIEW
46 PSE
47 XEQ 10
48 0
49 "├0"
50 AVIEW
51 PSE
52♦LBL 02
53 RCL 02
54 X<>Y
55 X>Y?
56 GTO 06
57 X<0?
58 GTO 06
59 " IGNITION"
60 AVIEW
61 PSE
62 ST- 02
63 RCL 02
64 10
65 X>Y?
66 XEQ 07
67 CLX
68 RDN
69 RDN
70 3
71 *
72 5
73 -
74 STO 03
75 2
76 /
77 RCL 00
78 +
79 RCL 01
80 +
81 RCL 03
82 ST+ 01
83 RDN
84 STO 00
85 INT
86 X>0?
87 GTO 01
88 -3
89 RCL 01
90 X<Y?
91 GTO 03
92 RCL 07
93 RCL 02
94 X>Y?
95 STO 07
96 X>Y?
97 XEQ 08
98 BEEP
99 XEQ 09
100 " TOUCH"
101 GTO 05
102♦LBL 03
103 XEQ 09
104 " CRASH"
105 GTO 05
106♦LBL 04
107 XEQ 09
108 " BURNT"
109♦LBL 05
110 "├ DOWN"
111 AVIEW
112 PSE
113 XEQ 10
114 "├E"
115 AVIEW
116 PSE
117 PSE
118 XEQ 08
119 PSE
120 1
121 ST+ 08
122 "GAME "
123 ARCL 08
124 "├ OVER"
125 AVIEW
126 PSE
127 GTO "AE"
128♦LBL 06
129 XEQ 09
130 " PILOT"
131 GTO 05
132♦LBL 07
133 "** DANGER **"
134 AVIEW
135 PSE
136 RTN
137♦LBL 08
138 "WITH "
139 ARCL 07
140 "├ BEST"
141 AVIEW
142 PSE
143 RTN
144♦LBL 09
145 CLA
146 ARCL 05
147 ARCL 06
148 AVIEW
149 PSE
150 RTN
151♦LBL 10
152 CLA
153 ARCL 00
154 ARCL 04
155 ARCL 01
156 ARCL 04
157 ARCL 02
158 ARCL 04
159 END
```

Harald M. Krumins (13)

****

C L U B B Ö R S E

Clubmitglieder können in der Clubbörse kostenlos Gelegenheitsanzeigen aufgeben !

Gegen 2,--DM Unkostenerstattung verleihe ich das Buch "Computer Chinesisch" von H.G.Klaus an Mitglieder.
Michael Kirchner (421), Goethestr. 29, 32oo Hildesheim

HP-41 Simulator für HP-1ooo: Das Prgr. läuft unter RTEIV/IVb, bietet alle 41-Funktionen und noch ein paar mehr und gibt ein komplettes Bild der Speicher etc. Ich stelle das Prgr. auf Band (HP-Minicartridge) oder Floppy-disk (8.5 inch, DD, DS) oder als Listing zur Verfügung.
Ernst Stelzer V (4oo), Gerhart-Hauptmann-Ring 127, 6ooo Frankfurt/M. 5o, Tel. o611/575556

Suche Kontakt zu HP-85 Benutzer / Besitzer !
Detlev Bock (oo2), Petrikirchstr. 36, 34oo Göttingen, Tel o551/33221

Moderne Taschenrechner- und Mathematik-Literatur, fast 7o Titel, neuwertig, mit 66,6 % unter Ladenpreis zu verkaufen. Verzeichnis anfordern mit 1,--DM frankiertem, adressierten Umschlag und 1,--DM in Briefmarken als Kostenbeitrag für Fotokopie des 3-seitigen Titel-Verzeichnis.
Gerhard Siewert (o52), Mozartstr. 14, 645o Hanau 1

Suche PPC-ROM zu kaufen !
Walter M. Schutz (242), Bahnhofstr. 43, CH-87o2 Zollikon

HP-25, kompl. mit Handbuch, PRGM-Sammlung, Netzladegerät, neuem Batteriesatz, Bestzustand, 85,-- zu verkaufen
HP-67, wie vor, für 35o,--DM zu verkaufen
Gerhard Siewert (o52), Mozartstr. 14, 645o Hanau 1, Tel. o6181/82464

Suche PPC-ROM zu kaufen !
Gerhard Schönemann (472), Kirchfeld 11, 3584 Zwesten, Tel. o5626/1458

Suche folgende Software: (1) Fluid Dynamics an Hydraulics (ooo41-9o139) - (2) Heating, Ventilating and Conditioning (ooo41-9o14o) - (3) Thermal and Transport Science (ooo41-15o19)
E. Barchewitz, Wasserwerkstr. 33, 68oo Mannheim 31

HP-41c & QUAD-RAM für VHB 65o,--DM zu verkaufen !
Olaf Pilling (268), Tel. o3o/4o15476

```
01+LBL "PZM
OD11"
02 FS? 55
03 GTO 05
04 GTO 06
05+LBL 05
06 "DIESES
PROGRAMM"
07 ACA
08 " ERRECH
-"
09 ACA
10 "NET FUE
R JEDE E"
11 ACA
12 "INGEGEB
-"
13 ACA
14 "ENE GAN
ZZAHLIGE"
15 ACA
16 " ZAHL D
IE"
17 ACA
18 "PRUEFZI
FFER NAC"
19 ACA
20 "H DEM M
O-"
21 ACA
22 "DULO 11
-VERFAHR"
23 ACA
24 "EN."
25 ACA
26 PRBUF
27+LBL A
28 CLX
29 CLRG
30 CLA
31 FIX 0
32 FS? 55
33 GTO 07
34 GTO 08
35+LBL 07
36 ADV
37 "BITTE Z
AHL EING"
38 ACA
39 "EBEN:"
40 ACA
41 PRBUF
42 STOP
43+LBL 09
44 STO 01
45 STO 02
46 0002,009
01
47 STO 03
48+LBL 01
49 RCL 02
50 ENTER↑
51 X=0?
52 GTO 03
53 1 E1
54 /
55 INT
56 STO 02
57 1 E1
58 *
59 -
60 RCL 03
61 INT
62 *
63 ST+ 04
64 ISG 03
65 GTO 01
66 GTO 02
67+LBL 02
68 0002,009
01
69 STO 03
70 GTO 01
71+LBL 03
72 RCL 04
73 11
74 MOD
75 11
76 X<>Y
77 -
78 10
79 X>Y?
80 GTO 04
81 0
82 X<>Y
83 GTO 04
84+LBL 04
85 X<>Y
86 STO 05
87 FS? 55
88 GTO 10
89 GTO 11
90+LBL 10
91 ADV
92 "DIE PRU
EFZIFFER"
93 ACA
94 " ZUR ZA
HL"
95 ACA
96 RCL 01
97 ACX
98 " NACH D
EM MODUL"
99 ACA
100 "O 11-VE
R"
101 ACA
102 "FAHREN
IST:"
103 ACA
104 RCL 05
105 ACX
106 "."
107 ACA
108 PRBUF
109 GTO A
110+LBL 06
111 "PRUEFZI
FFERN PR"
112 "-OGRAMM
"
113 AVIEW
114 GTO A
115+LBL 08
116 "WELCHE
ZAHL?"
117 PROMPT
118 GTO 09
119+LBL 11
120 "P-ZIFFE
R="
121 ARCL X
122 AVIEW
123 GTO A
124 END
```

<u>Prüfzifferniprogramm PZMOD11</u>

Wegen seiner zunehmenden Bedeutung soll nachfolgend die Form der Eingabesicherung durch eine Nummernprüfung kurz erläutert werden. Durch Prüfziffern werden vornehmlich Ordnungsbegriffe gegen Fehler bei der Datenerfassung (Schreib - Lesefehler) abgesichert. Die jeweils an den Ordnungsbegriff angefügte Prüfziffer wird bei der Eingabe automatisch mit einer errechneten Prüfziffer verglichen. Bei Ungleichheit erfolgt eine Fehlermeldung. Die Prüfziffer ist somit ein einstelliges Zeichen, das an den Ordnungsbegriff angefügt wird, um diesen sich selbst prüfend zu gestalten.
In der Datenverarbeitung, vornehmlich in der EDV-Organisation, sind verschiedene Prüfverfahren bekannt, um solche Prüfziffern zu errechnen. Die einfachsten Verfahren bedienen sich der Quersummenrechnung, was selbstverständlich zur Folge hat, dass Zahlenvertauschungen (Zahlendreher) nicht erkannt werden. Anspruchsvollere Verfahren bedienen sich sogenannter Moduli. Der Sortierbegriff wird mit einer bestimmten Zahlenfolge multipliziert, addiert und danach die Summe der Produkte durch den Modulus dividiert. Der sich dabei bildende Rest wird vom Modulus subtrahiert und bildet die Prüfziffer. Die hierbei bekanntesten Verfahren sind die Modulo-10-Verfahren, bzw. das am häufigsten Angewandte, das Modulo-11-Verfahren. Dieses Verfahren wird von allen bekannten Prüfgeräteherstellern angewandt und ist in der Praxis sehr weit verbreitet. Beispielsweise werden von diesem Verfahren folgende Fehler zu 100% erkannt: Schreibfehler, einfache, doppelte und sonstige Drehfehler.

<u>Wie arbeitet nun das Modulo-11-Verfahren ?</u>

1. Jede Stelle des Ordnugsbegriffes wird von rechts nach links mit 2,3,4,...9 multipliziert
2. Die so ermittelten Produkte werden addiert
3. Die Summe dieser Produkte wird durch 11 dividiert
4. Der verbleibende Rest wird von 11 subtrahiert, das Ergebnis ist die Prüfziffer
5. Bei langen Zahlen wiederholt sich die Gewichtung
6. Bei Rest 0 und 1 ergibt sich ein zweistelliges Zeichen, die durch das Verfahren nicht verarbeitet werden können, und in die Prüfziffer 0 umgewandelt werden, der sich ergebende Fehler ist vernachlässigbar.

Das beigefügte Programm wurde so aufgebaut, dass es mit oder ohne Drucker angewendet werden kann. Ueber das Drucker-Anwesenheitsflag entscheidet das Programm selbstständig, ob gedruckt werden muss oder nicht.

Hartmut Brücker (229)

```
01+LBL "PZM
OD11"
02 CF 29
03+LBL A
04 0
05 STO 04
06 "ZAHL ?"
07 PROMPT
08 STO 01
09 STO 02
10 2,009
11 STO 03
12+LBL 01
13 RCL 02
14 ENTER↑
15 X=0?
16 GTO 03
17 10
18 /
19 INT
20 STO 02
21 10
22 *
23 -
24 RCL 03
25 INT
26 *
27 ST+ 04
28 ISG 03
29 GTO 01
30 8
31 ST- 03
32 GTO 01
33+LBL 03
34 RCL 04
35 11
36 MOD
37 LASTX
38 X<>Y
39 -
40 10
41 X>Y?
42 GTO 04
43 0
44 X<>Y
45+LBL 04
46 X<>Y
47 STO 05
48 "P-ZIFFE
R= "
49 FIX 0
50 ARCL X
51 AVIEW
52 PSE
53 GTO A
54 END
```

Soweit die Programmbeschreibung von Hartmut Brücker. Ich habe dieses Programm von Oliver erhalten und soweit bearbeitet, dass es abgedruckt werden kann. Hartmut lässt sich das Programm selbst erklären, was es tut. Ich fasse dies als eine Art Programmdokumentation auf, die jedoch ein HP-45C-Programm nur unübersichtlich macht. Im Übrigen kann die Programmausführung mit oder ohne Drucker auch einfacher erreicht werden über den AVIEW-Befehl. Dass die gesamten Datenspeicher, das X- und ALPHA-Register gelöscht werden ist nicht nötig, denn die Speicherinhalte werden durch eine neue Abspeicherung überschrieben. Als einziges Register muss das Datenregister Ø4 gelöscht werden, da darin nur aufsummiert wird. Durch die generelle Löschung wird auch die letzte Abspeicherung der Prüfzahl (Schritt 86) in das Register Ø5 sinnlos. Am Anfang des Programms würde ich noch das Flag 29 löschen, denn zur Prüfzifferndarstellung wird der Punkt (oder das Komma) nicht benötigt. Dies ist jedoch eine rein optische Angelegenheit.
In Schritt 46 und 68 müssen die Vornullen nicht geschrieben werden, da sie der Rechner bei der Programmausführung sowieso unterdrückt (kann jeder im RUN-Modus selbst ausprobieren). Auch die Inkrement-Schrittweite Ø1 muss nicht eingegeben werden, da der Rechner beim Fehlen dieser Angabe automatisch um 1 in- oder dekrementiert.
Durch eine geschicktere Abfrage des Flags 55 in der Programmzeile 32 könnten einige Bytes an Programmspeicherplatz gespart werden: FC? 55 ; GTO Ø8 . Das Label Ø7 und der vorherige Sprung (GTO Ø7) sind nun überflüssig geworden. Ebenso wird bei der Abfrage des Flags 55 im Programmschritt 87 auf dieselbe Weise das Label 1Ø und der Sprung GTO 1Ø überflüssig.
In den Schritten 53 und 57 verwendet Hartmut die Zahl 1Ø, schreibt sie jedoch etwas umständlich mit 1 E1 und verschenkt dadurch ein Byte im Programmspeicher.
Das Label Ø2 in Schritt 67 ist nicht nötig, denn die Ausführungsreihenfolge der Befehle würde sowieso nacheinander ausgefürt werden, genau gleich, wie bei der Befehlsfolge GTO Ø2 ; LBL Ø2 . Ebenso wird bei Schritt 83 das GTO Ø4 unnötig, jedoch das darauffolgende Label Ø4 nicht, da es vom Schritt 8Ø aus angesprungen wird. In den Programmschritten 68 und 69 kann das Register Ø3 mit 3 statt 6 Bytes wieder hergestellt werden: da darin der Wert 1Ø,ØØ9 stehen muss, wenn der Rechner hier "vorbeikommt" kann 8 ; ST- Ø3 geschrieben werden. Wenn man noch in Schritt 75 den Befehl LASTX verwendet, ist schon wieder ein Byte gespart. Dieses Programm habe ich so von 447 auf 1Ø3 Bytes gekürzt (Vergleich linke - rechts).

Happy Programming

Fred Huber (186)

```
01+LBL "LOG      93 ARCL 14      186 INT         232 TONE 9      276 X<> IND     311 STO IND
IC"              94 PROMPT       187 1           233+LBL 14      13              14
02 ΣREG 01       95 SF 02        188 -           234 CLD         277 X≠0?        312 X<> IND
03 CF 04         96 FC?C 00      189 1 E3        235 AVIEW       278 GTO 00      13
04 CF 05         97 GTO 17       190 /           236 PSE         279 X<> IND     313 RDN
05+LBL 15        98+LBL 04       191 STO 14      237 FC? 22      13              314 STO 13
06 FIX 0         99 CF 02        192 ISG 14      238 GTO 14      280 X<>Y        315 X<> IND
07 CF 00         100 CF 04       193+LBL 10      239 X>Y?        281 STO 13      13
08 CF 01         101 "END"       194 RCL IND     240 GTO 13      282 X<> IND     316 ST* IND
09 CF 02         102 PROMPT      14              241 STO IND     13              14
10 CF 03         103 GTO 04      195 X≠0?        Z               283 X=0?        317 X<> IND
11 CF 06         104+LBL 05      196 GTO 11      242 ST+ 18      284 GTO 00      13
12 CF 22         105 FS?C 01     197 1           243 ISG Z       285 X<> IND     318 RDN
13 SF 26         106 GTO 06      198 STO IND     244 GTO 13      13              319 GTO 19
14 CF 29         107 "WAHR"      14              245 XEQ 08      286 RDN         320+LBL "HAN
15 CLRG          108 BEEP        199 ST+ 15      246 GTO 16      287 CLX         D"
16 12            109 GTO 00      200 ISG 14      247+LBL "XOR    288 GTO 18      321 XEQ "AND
17 "ELEMENT      110+LBL 06      201 GTO 10      "               289+LBL 00      "
E ?"             111 "UNLOESB    202+LBL 11      248 SF 03       290 X<> IND     322+LBL "NON
18 TONE 9        AR"             203 1           249+LBL "EQU    13              "
19+LBL 01        112 TONE 2      204 ST- 13      I"              291 +           323 STO 13
20 CLD           113 TONE 2      205 CLX         250 STO 13      292 GTO 18      324 X<> IND
21 AVIEW         114+LBL 00      206 STO IND     251 X<> IND     293+LBL "OR"    13
22 PSE           115 PROMPT      13              13              294 STO 13      325 X≠0?
23 FC? 22        116 FS?C 00     207 DSE 15      252 STO IND     295 X<> IND     326 GTO 00
24 GTO 01        117 GTO 04      208 GTO 11      13              13              327 X<> IND
25 X≠0?          118 SF 02       209+LBL 12      253 X<>Y        296 STO IND     13
26 X>Y?          119+LBL 17      210 CF 03       254 STO 13      14              328 GTO 18
27 GTO 15        120 FS?C 05     211 DSE 17      255 X<> IND     297 X<> IND     329+LBL 00
28 STO 00        121 GTO 05      212 RTN         13              13              330 X<> IND
29 STO 18        122 FS? 00      213 CF 02       256 STO IND     298 RDN         13
30 1 E3          123 GTO 05      214 DSE 18      13              299 STO 13      331 CLX
31 /             124 1.9         215 GTO 17      257 X>Y?        300 X<> IND     332+LBL 18
32 1             125 STO 13      216 CLX         258 X<>Y        13              333 STO IND
33 +             126 FS? 02      217 STO IND     259 X≠0?        301 ST+ IND     14
34 STO 16        127 GTO 00      00              260 GTO 00      14              334 RDN
35 FS? 05        128 CLΣ         218 RTN         261 X<>Y        302 X<> IND     335+LBL 19
36 RTN           129 CLX         219+LBL "PS"    262 X=0?        13              336 RCL 14
37 GTO 17        130 STO 07      220 SF 04       263 GTO 00      303 RDN         337 ISG 14
38+LBL 16        131 STO 08      221 SF 05       264 +           304 GTO 19      338 RTN
39 19.9          132 STO 09      222 XEQ 15      265 FC?C 03     305+LBL "NOR    339+LBL "Σ"
40 STO 14        133 STO 10      223 1           266 CLX         "               340 SF 01
41 RCL 16        134 STO 11      224 RCL 01      267 GTO 18      306 XEQ "OR"    341 19.9
42 STO 15        135 STO 12      225 STO 18      268+LBL 00      307 GTO "NON    342 STO 14
43 CLA           136 RCL 18      226+LBL 13      269 X<> 13      "               343 RCL IND
44+LBL 02        137 STO 15      227 RDN         270 +           308+LBL "AND    Y
45 ARCL IND      138 1           228 CLA         271 FS?C 03     "               344 ST* 15
15               139+LBL 07      229 ARCL Y      272 CLX         309 STO 13      345 X=0?
46 RCL IND       140 STO IND     230 ". WERT     273 GTO 18      310 X<> IND     346 GTO 17
15               15              ?"              274+LBL "IF"    13              347 END
47 X≠0?          141 DSE 15      231 CF 22       275 STO 13
48 SF 03         142 GTO 07
49 X≠0?          143+LBL 00
50 GTO 00        144 XEQ "PM"
51 FS? 03        145 SF 02
52 SF 06         146 GTO 16
53+LBL 00        147+LBL "PM"
54 ISG 15        148 FS? 02
55 GTO 02        149 GTO 09
56 FC?C 03       150+LBL 08
57 SF 00         151 RCL 00
58 1             152 FACT
59 FC? 04        153 RCL 18
60 RCL 17        154 FACT
61 FC?C 06       155 /
62 STO 17        156 RCL 00
63 AVIEW         157 RCL 18
64 TONE 5        158 -
65 XEQ "P"       159 FACT
66 ASTO 13       160 /
67 ASHF          161 STO 17
68 ASTO 14       162 RTN
69 FS?C 01       163+LBL 09
70 GTO 03        164 RCL IND
71 RCL IND       13
X                165 ISG 13
72 X≠0?          166 RCL IND
73 GTO 17        13
74 SF 03         167 X<Y?
75 "FALSCH:      168 GTO 00
"                169 X≠Y?
76 TONE 2        170 SF 03
77 TONE 2        171 RCL 00
78 GTO 00        172 RCL 13
79+LBL 03        173 X<Y?
80 RCL 15        174 GTO 09
81 X=0?          175 GTO 12
82 GTO 17        176+LBL 00
83 "LOESUNG      177 1
:"               178 STO IND
84 BEEP          13
85+LBL 00        179 ST- 13
86 CF 05         180 CLX
87 AVIEW         181 STO IND
88 PSE           13
89 CLA           182 STO 15
90 FS?C 03       183 FC?C 03
91 "----"        184 GTO 12
92 ARCL 13       185 RCL 13
```

L O G I C

Das Programm dient zur Prüfung logischer Gesetze (True/False) und zur Lösung von LOGICALS der Art, deren Bekanntestes mit der Frage "Wem gehört das Zebra?" schliesst.

I. Prüfen logischer Gesetze

1. Eingabe des logischen Gesetzes als Programm "P" (Siehe Beispiel)
2. Initialisation des Hauptprogrammes durch XEQ'LOGIC
3. Auf die Frage "ELEMENTS ?" wird die Anzahl der Variablen des Gesetzes eingegeben
4. Darauf permutiert der Rechner die möglichen Wahrheitsverteilungen auf die n Elemente durch und zeigt an:
   a) "TRUE" wenn das Gesetz gültig ist
   b) "FALSE" wenn die Wahrheitswertbelegung zu einem falschen Resultat führt, diese Belegung wird anschliessend angezeigt, z.B.: ---OIOOIO
5. Im Fall 4a führt R/S zur Anzeige "END", da keine weitere Permutation der Wahrheitswerte mehr möglich ist
6. Im Fall 4b permutiert der Rechner die restlichen Wahrheitswertbelegungen durch (falls dies nicht bereits die letzte Permutation war)
7. Will man die Befragung einer ganz bestimmten Wahrheitswertbelegung durchführen, so wird die Abfrage mit XEQ'PS initialisiert, der Rechner fragt dann nach der Anzahl der Elemente und darauf nach dem Wahrheitswert (O,I) der einzelnen Elemente: "x. VALUE" . Nach dem letzten Wahrheitswert untersucht der Rechner die Belegung und weist das Ergebnis aus: "TRUE" oder "FALSE" . Für die Fortführung der Abfrage mit permutierter Wahrheitswertverteilung gehe zu 5.

II. Lösen eines LOGICALS

1. 'Intelligente' Eingabe der Aussagen des Logicals als Programm "P" (Siehe Beispiel)
2. Initialisation des Hauptprogramms durch XEQ'LOGIC
3. Auf die Frage "ELEMENTS ?" wird die Anzahl der Variablen des Logicals eingegeben
4. Darauf permutiert der Rechner die möglichen Wahrheitswertverteilungen (Werte: O,I) durch und zeigt an:
   a) "UNSOLVABLE" wenn das Logical keine Lösung hat
   b) "SOLUTION" und anschliessend die Wahrheitswertbelegung z.B. "OOIOOIOIO", die zur Lösung führte, wenn die erste (nicht unbedingt die einzige!) Belegung zu einem Ergebnis führt

5. Eventuelle Befragung der restlichen Permutationen mittels R/S. Dies führt allerdings zur Anzeige "END", falls es keine weiteren Permutationen gibt.
6. Will man die Befragung einer ganz bestimmten Wahrheitswertbelegung durchführen, so wird die Abfrage mit XEQ'PS gestartet, der Rechner fragt dann nach der Anzahl der Elemente und darauf nach dem Wahrheitswert (O,I) der einzelnen Elemente: "x. VALUE". Nach dem letzten Wahrheitwert untersucht der Rechner die Belegung und weist das Ergebnis aus: "SOLUTION" oder "UNSOLVABLE". Für die Fortführung der Abfrage mit permutierter Wahrheitswertverteilung gehe zu 5.

Beispiele

I. Prüfen eines logischen Gesetzes

1. Beispiel:

$\vdash (A \wedge B \rightarrow C) \overset{?}{\leftrightarrow} (B \wedge A \rightarrow C)$ mit A=1 , B=2 , C=3

Eingabe des Problems als Programm "P":

```
LBL'P
1
ENTER↑
2
XEQ'AND
3
XEQ IF
2
ENTER↑
XEQ'AND
3
XEQ'IF
XEQ'EQUI
```

Die Formel enthält 3 Elemente (A, B, C).
Also nach XEQ'LOGIC wird auf die Frage "ELEMENTS ?" die Zahl 3 eingegeben.
Der Rechner permutiert nun sämtliche möglichen Wahrheitswertbelegungen (I= wahr, O= falsch) durch. Findet er keine Belegung, bei der das zu prüfende Gesetz falsch wäre, kündigt er dies durch BEEP und "TRUE" an.

2. Beispiel:

$\vdash \phi \wedge \psi \overset{?}{\leftrightarrow} \neg(\neg \phi \wedge \neg \psi)$ mit 2 Elementen: ϕ=1 , ψ=2

```
LBL'P
1
ENTER↑
2
XEQ'AND
1
XEQ'NON
2
XEQ'NON
XEQ'AND
XEQ'NON
XEQ'EQUI
```

Bei der Belegung: IO (d.h. ϕ=wahr, ψ=falsch) ergibt sich ein Widerspruch Anzeige: ---IO. Richtig wäre das Gesetz als: $\vdash \phi \vee \psi \leftrightarrow \neg(\neg \phi \wedge \neg \psi)$

Frage nach einer bestimmten Wahrheitsbelegung, z.B. OI, Start des Programms mit XEQ'PS, der Rechner fragt dann nach der Anzahl der Elemente, Eingabe: 2, danach fragt der Rechner nach dem ersten Wahrheitswert: hier O, dann nach dem zweiten und letzten Wert: hier 1. Dann untersucht der Rechner diese Belegung, die hier zu der Anzeige "FALSE" und "---OI" führt. Wollte man die nächste Permutation prüfen (hier OO), so drückt man R/S; dies führt dann zur Anzeige: "TRUE".

II. Lösen eines Logicals

(Quelle: 99 Logeleien, Zweistein)
Einige Familienmitglieder (alle?) der Meyers werden zu Besuch erwartet. Folgendes ist gewiss:
1) Wenn Vater Meyer kommt, dann auch Frau Meyer
2) Mindestens einer der beiden Söhne Uwe und Kay kommt
3) Entweder kommt Frau Meyer oder Tim (der dritte Sohn)
4) Entweder kommen Tim und Kay oder beide nicht
5) Wenn Uwe kommt, dann auch Kay und Vater Meyer
Wer kommt ?

Sei Vater=1, Mutter=2, Kay=3, Uwe=4, Tim=5; also 5 Elemente.
Dann lassen sich die 'Gleichungen' 1) bis 5) auch verkürzt folgendermassen schreiben:
1) $1 \rightarrow 2$ (Subjunktion IF)
2) $4 \vee 3$ (Disjunktion OR)
3) $2 \succ\prec 5$ (Alternative XOR)
4) $(5 \wedge 3) \succ\prec (\neg 5 \wedge \neg 3)$ (Negation NON)
5) $4 \rightarrow (3 \wedge 1)$ (Konjunktion AND)

Verständlicherweise müssen die fünf Aussagen noch miteinander verknüpft werden, da sie ja als Gesamtheit Gültigkeit haben. Dies geschieht mit der Σ-Funktion.
Demnach sieht das Programm "P" nunmehr so aus:

```
LBL'P       3
1           XEQ'NON
ENTER↑      XEQ'AND
2           XEQ'XOR
XEQ'IF      XEQ'Σ
XEQ'Σ       4
4           ENTER↑
ENTER↑      3
3           ENTER↑
XEQ'OR      1
XEQ'Σ       XEQ'AND
2           XEQ'IF
ENTER↑      XEQ'Σ
5
XEQ'XOR
XEQ'Σ
5
ENTER↑
3
XEQ'AND
5
XEQ'NON
```

Nun folgt der Programmstart mit XEQ'LOGIC, auf die Frage "ELEMENTS ?" wird die Anzahl der Elemente eingegeben: 5. Dann permutiert der Rechner die verschiedenen Personengruppierungen durch, bis er eine Kombination findet, die zur Gesamtheit der Aussagen 1) bis 5) keinen Widerspruch enthält und weist diese Kombination als "SOLUTION" aus, hier OOIOI, was in diesem Beispiel folgendes bedeutet: nur Person 3 und Person 5 kommen zu Besuch, mithin Kay und Tim.
Lässt man den Rechner weiter permutieren mittels R/S, so findet er keine weitere "SOLUTION" mehr (die Lösung ist also eindeutig !) und weist für die restlichen Permutationen "UNSOLVABLE" aus

Thomas Lindner (159)

A U T O M O B I L K O S T E N

Die Programme AA bis AD dienen der Kontrolle der bei einem Automobil anfallenden Kosten. Die in den Programmen gebrauchten Abkürzungen basieren auf der englischen Sprache und können vom Anwender beliebig abgeändert und modifiziert werden.
Voraussetzung für ein korrektes Arbeiten der Programme AA, AB und AD ist es, dass der Wagen vollgetankt wird und der Kilometerstand im Datenspeicher ØØ abgelegt wird. Danach ist jeweils wieder vollzutanken und der Kilometerstand ist auf dem Kassenbon zu vermerken. Der Preis und die bezogene Benzinmenge stehen ja ohnehin schon darauf. Mit diesem Bon kann die Rechnung entsprechend der untenstehenden Bedienungsanleitungen begonnen werden. Bei den anderen Kosten in den Programmen AA und AC ist nur deren Eingabe erforderlich.
Das Programm CARCOST (AA) ist eine Kombination der Programme FUELCOST (AB) und TOTAL (AC). CARCOST ist im Handling angenehmer. FUELCOST und TOTAL belegen aber, einzeln abgespeichert, erheblich weniger Speicherplatz. Deshalb sind sie als modifizierte Einzelprogramme beigefügt. Das Programm FUELCOST EXCESS (AD) ist für diejenigen Autobesitzer gedacht, die es ganz genau wissen wollen.

Bedienunganleitungen

CARCOST (AA)

SIZE Ø11
PRGM einlesen
CLRG
XEQ'AA
"CARCOST"
"CARD"
Datenkarte einlesen
oder 3x R/S
"B OR D"
wähle B
"FUELC"
"PRT ?"
Preis eingeben
z.B. 12.58 DM
R/S
"LIT ?"
Liter eingeben
z.B. 10,5 l
R/S
"KMN ?"
neuer km-Stand eingeben
z.B. 200 km
R/S

Kontrollanzeigen:
"KMO Ø.ØØ"
alter km-Stand
"KMN 2ØØ.ØØ"
neuer km-Stand
"KMD 2ØØ.ØØ"
gefahrene km
"LIT 1Ø.5Ø"
getankte Liter
"LIH 5.25"
Verbrauch pro 100 km
"PRT 12.58"
Preis beim Tanken
"PRL 1.2Ø"
Preis pro Liter
"PRH 6.29"
Preis pro 100 km
"FCT 12.58"
Gesamtpreis sämtlicher Tankvorgänge
"CARD"
Datenkarte einlesen
oder 3x R/S
"CARCOST"
weiter wie oben

wähle D
"TOTAL"
"FCT"
siehe B
"INS"
"INS NEW ?"
"INS"
Versicherung
"TAX"
"TAX NEW ?"
"TAX"
Steuern
"REP"
"REP NEW ?"
"REP"
Reparaturen
"DIV"
"DIV NEW ?"
"DIV"
Diverse Kleinteile (Oel, Filter, etc)
"EQU"
"EQU NEW ?"
"EQU"
Ausstattung
"GRAND"
Gesamtsumme sämtlicher Kosten
"CARD"
Datenkarte einlesen
oder 3x R/S
"CARCOST"
weiter wie oben

Während der Anzeige z.B. "INS NEW ?" kann durch R/S das Programm unterbrochen werden und der entsprechende Wert eingegeben werden, dann wieder R/S.
Beispiel:
"INS"
"INS NEW ?"
R/S
Eingabe des Wertes
z.B. 250.- DM
R/S
"INS 25Ø.ØØ"

Ohne Eingabe läuft das Programm selbständig weiter.

FUELCOST (AB)

SIZE ØØ2
PRGM einlesen
CLRG
XEQ AB
"FUELC"
Bedienung wie unter CARCOST B

TOTAL (AC)

SIZE Ø1Ø
PRGM einlesen
CLRG
XEQ AC
"TOTAL"
"TOTAL Ø.ØØ"
Bedienug wie unter CARCOST D

FUELCOST EXCESS (AD)

SIZE Ø14
PRGM einlesen
CLRG
XEQ AD
"FUELC EXCESS"
R/S
"DATA CARD"
Datenkarte einlesen
oder 3x R/S
"PRT ?"
Preis eingeben
z.B. 12.58 DM
R/S
"LIT ?"
Liter eingeben
z.B. 1o,5 l
R/S
"KMN ?"
neuer km-Stand eingeben
z.B. 200 km
R/S
"CHECK INPUT"
"Ø.ØØ"
"12.58"
"1Ø.5Ø"
"2ØØ.ØØ"
R/S
"FOR FUELING"
"KMO Ø.ØØ"
alter km-Stand
"KMN 2ØØ.ØØ"
neuer km-Stand
"KMD 2ØØ.ØØ"
gefahrene km
"LIT 1Ø.5Ø"
getankte Liter
"LIH 5,25"
Verbrauch pro 100 km
"PRT 12.58"
Preis beim Tanken
"PRL 1.2Ø"
Preis pro Liter
"PRH 6.29"
Preis pro 100 km

"FOR MONTH"
Berechnung für Monat
"KMD "
"LIT "
"LIH "
"PRT "
"PRL "
"PRH "

"FOR YEAR"
Berechnung für Jahr
"KMD "
"LIT "
"LIH "
"PRT "
"PRL "
"PRH "

"TOTAL TIME"
Gesamte Nutzungsdauer des Fahrzeuges
"KMD "
"LIT "
"LIH "
"PRT "
"PRL "
"PRH "

"DATA CARD"
Datenkarte einlesen
oder 3x R/S
"FUELC EXCESS"
weiter wie oben

Das Programm sieht den Sprung des Tachometers von 99999 auf 00000 nicht vor, das heisst in diesem Falle, dass der neue km-Stand kleiner als der alte km-Stand ist, und der Rechner wieder zur Abfrage zurückspringt. Es wird dabei keine Speicherung der Daten vorgenommen.

Für die Beendigung einer Rechnungsperiode sind die entsprechenden Datenspeicher zu löschen.

Harald M. Krumins (13)

```
01•LBL "AA"
02 FIX 2
03 "CARCOST
"
04 XEQ 02
05 XEQ 07
06 RDTA
07 SF 27
08 "    B OR
D"
09 AVIEW
10 STOP
11•LBL B
12 CF 27
13 "FUELC"
14 XEQ 02
15 CLST
16 "PRT ?"
17 PROMPT
18 "LIT ?"
19 PROMPT
20 "KMN ?"
21 PROMPT
22 RCL 00
23 "KMO "
24 XEQ 01
25 X<>Y
26 X<=Y?
27 GTO B
28 X<>Y
29 RDN
30 "KMN "
31 XEQ 01
32 STO 00
33 R↑
34 -
35 "KMD "
36 XEQ 01
37 100
38 /
39 RDN
40 "LIT "
41 XEQ 01
42 STO Z
43 R↑
44 /
45 "LIH "
46 XEQ 01
47 RDN
48 ST+ 08
49 ST+ 06
50 "PRT "
51 XEQ 01
52 X<>Y
53 /
54 "PRL "
55 XEQ 01
56 R↑
57 *
58 "PRH "
59 XEQ 01
60 XEQ 06
61 XEQ 07
62 WDTA
63 GTO "AA"
64•LBL D
65 CF 27
66 "TOTAL"
67 XEQ 02
68 "NEW  ?"
69 ASTO 09
70 CLST
71 STO 10
72 XEQ 06
73 "INS "
74 XEQ 05
75 "TAX "
76 XEQ 05
77 "REP "
78 XEQ 05
79 "DIV "
80 XEQ 05
81 "EQU "
82 XEQ 05
83 RCL 08
84 "GRAND "
85 XEQ 01
86 XEQ 07
87 WDTA
88 GTO "AA"
89•LBL 03
90 RCL IND
10
91 ARCL 07
92•LBL 01
93 ARCL X
94•LBL 02
95 AVIEW
96 PSE
97 PSE
98 CLA
99 RTN
100•LBL 04
101 ARCL 07
102 ARCL 09
103 XEQ 02
104 ST+ IND
10
105 ST+ 08
106 CLA
107 RTN
108•LBL 05
109 ASTO 07
110 CLA
111 1
112 ST+ 10
113 XEQ 03
114 CLX
115 XEQ 04
116 XEQ 03
117 CLX
118 RTN
119•LBL 06
120 RCL 06
121 "FCT "
122 XEQ 01
123 RTN
124•LBL 07
125 CLST
126 SF 14
127 "CARD"
128 AVIEW
129 END
```

```
01•LBL "AB"
02 "FUELC "
03 AVIEW
04 PSE
05 FIX 2
06 RCL 01
07 XEQ 01
08 STOP
09 CLST
10 SF 14
11 RDTA
12 "PRT ?"
13 PROMPT
14 "LIT ?"
15 PROMPT
16 "KMN ?"
17 PROMPT
18 RCL 00
19 "KMO "
20 XEQ 01
21 RDN
22 "KMN "
23 XEQ 01
24 STO 00
25 R↑
26 -
27 "KMD "
28 XEQ 01
29 100
30 /
31 RDN
32 "LIT "
33 XEQ 01
34 STO Z
35 R↑
36 /
37 "LIH "
38 XEQ 01
39 RDN
40 ST+ 01
41 "PRT "
42 XEQ 01
43 X<>Y
44 /
45 "PRL "
46 XEQ 01
47 R↑
48 *
49 "PRH "
50 XEQ 01
51 "CARD"
52 AVIEW
53 CLST
54 SF 14
55 WDTA
56 GTO "AB"
57•LBL 01
58 ARCL X
59 AVIEW
60 PSE
61 PSE
62 END
```

```
01•LBL "AC"
02 "TOTAL "
03 AVIEW
04 PSE
05 FIX 2
06 RCL 08
07 XEQ 01
08 STOP
09 CLST
10 SF 14
11 RDTA
12 "NEW ?"
13 ASTO 09
14 CLST
15 STO 00
16 "INS  "
17 XEQ 05
18 "TAX  "
19 XEQ 05
20 "REP  "
21 XEQ 05
22 "DIV  "
23 XEQ 05
24 "EQU  "
25 XEQ 05
26 "FCT  "
27 XEQ 05
28 CLST
29 SF 14
30 WDTA
31 GTO "AC"
32•LBL 03
33 RCL IND
00
34 ARCL 07
35•LBL 01
36 ARCL X
37•LBL 02
38 AVIEW
39 PSE
40 PSE
41 CLA
42 RTN
43•LBL 04
44 ARCL 07
45 ARCL 09
46 XEQ 02
47 ST+ IND
00
48 ST+ 08
49 CLA
50 RTN
51•LBL 05
52 ASTO 07
53 CLA
54 1
55 ST+ 00
56 XEQ 03
57 CLX
58 XEQ 04
59 XEQ 03
60 CLX
61 END
```

```
01•LBL "AD"
02 "FUELC E
XCESS"
03 AVIEW
04 STOP
05 XEQ 04
06 RDTA
07•LBL 05
08 CLST
09 FIX 2
10 "PRT ?"
11 PROMPT
12 "LIT ?"
13 PROMPT
14 "KMN ?"
15 PROMPT
16 "CHECK
INPUT"
17 XEQ 03
18 7PRSTK
19 STOP
20 "FOR  FU
ELING"
21 XEQ 03
22 RCL 00
23 "KMO "
24 XEQ 02
25 X<>Y
26 X<=Y?
27 GTO 05
28 STO 00
29 "KMN "
30 XEQ 02
31 X<>Y
32 -
33 STO 01
34 ST+ 04
35 ST+ 07
36 ST+ 10
37 RDN
38 STO 02
39 ST+ 05
40 ST+ 08
41 ST+ 11
42 RDN
43 STO 03
44 ST+ 06
45 ST+ 09
46 ST+ 12
47 CLST
48 STO 13
49 XEQ 00
50 " FOR  M
ONTH"
51 XEQ 03
52 XEQ 00
53 "  FOR Y
EAR"
54 XEQ 03
55 XEQ 00
56 " TOTAL
TIME"
57 XEQ 03
58 XEQ 00
59 XEQ 04
60 WDTA
61 GTO "AD"
62•LBL 00
63 CLST
64 XEQ 01
65 "KMD "
66 XEQ 02
67 100
68 /
69 XEQ 01
70 STO Z
71 "LIT "
72 XEQ 02
73 X<>Y
74 /
75 STO Z
76 "LIH "
77 XEQ 02
78 XEQ 01
79 "PRT "
80 XEQ 02
81 RCL Z
82 /
83 "PRL "
84 XEQ 02
85 *
86 "PRH "
87 XEQ 02
88 RTN
89•LBL 01
90 1
91 ST+ 13
92 RDN
93 RCL IND
13
94 RTN
95•LBL 02
96 ARCL X
97•LBL 03
98 AVIEW
99 PSE
100 PSE
101 RTN
102•LBL 04
103 CLST
104 SF 14
105 " DATA
CARD"
106 AVIEW
107 END
```

C L U B B Ö R S E

Suche PPC-ROM zu kaufen ! Hasso Schumacher (443), Tel. 040/6724110

Suche PPC-ROM zu kaufen ! Michael Haase (285), Tel. 07305/7736

O S B O R N E   1

Ab Ende dieses Jahres wird die Mitglieder- und Kontenverwaltung des Clubs über EDV laufen, da die Arbeit anders nicht mehr zu bewältigen ist. Dazu ist bereits ein Personalcomputer OSBORNE 1 bestellt. Dieser ab Frühjahr 82 auch in Deutschland in größeren Stückzahlen erhältliche Rechner ist um den Faktor drei billiger als alle derzeitigen Personalcomputer. Das Komplettsystem enthält Standardsoftware, die allein nach heutigen Preisen über 1500 $ kostet. Trotzdem liegt der Preis für 64k-Rechner, CP/M, 2 Disketten, 128 x 24 Monitor, und vieles mehr <u>unter</u> 1800 $. Der Rechner ist aktenkoffergroß (mit Monitor, Disk.laufw.) und transportabel. Über Batterien oder Akku läuft er ohne Netzanschluß bis zu 5 Stunden ! Oliver(1)

AOS für UPN-Rechner

Dieses Programm simuliert das AOS-System (bekanntester Vertreter: Texas Instruments). Die Bedienung erfolgt in ähnlicher Weise wie bei einem TI-Taschenrechner.

Es ist zweckmässig folgende Tastenzuordnungen vorzunehmen:

ᵀ+ → +	ᵀYX → Shift Y^X	ᵀCLR → Shift CLX
ᵀ- → -	ᵀINV → X<>Y	ᵀ= → ENTER↑
ᵀ* → *	ᵀ< → Shift LBL	
ᵀ/ → /	ᵀ> → Shift GTO	

Man kann im USER-Modus mit dem AOS-System, ohne USER-Modus mit dem UPN-System arbeiten.

Die Zahl, die angezeigt wird, steht im X-Register und kann daher mit allen 'einstelligen' Operationen (wie SIN, COS, TAN, LOG, STO..,1/X, usw) bearbeitet werden, auch während einer Rechnung. Die Dateneingabe kann auch durch RCL.., usw erfolgen. Der Anzeigemodus kann frei gewählt werden.

In diesem Programm ist die Anzahl der Klammerebenen von der Anzahl der verfügbaren Datenspeicherregister abhängig. Die Anzahl der möglichen Klammerebenen kann nach folgender Formel berechnet werden:

$$\text{Klammerebenen} = \text{INT}\left(\frac{\text{Datenspeicher} - 2}{4}\right)$$

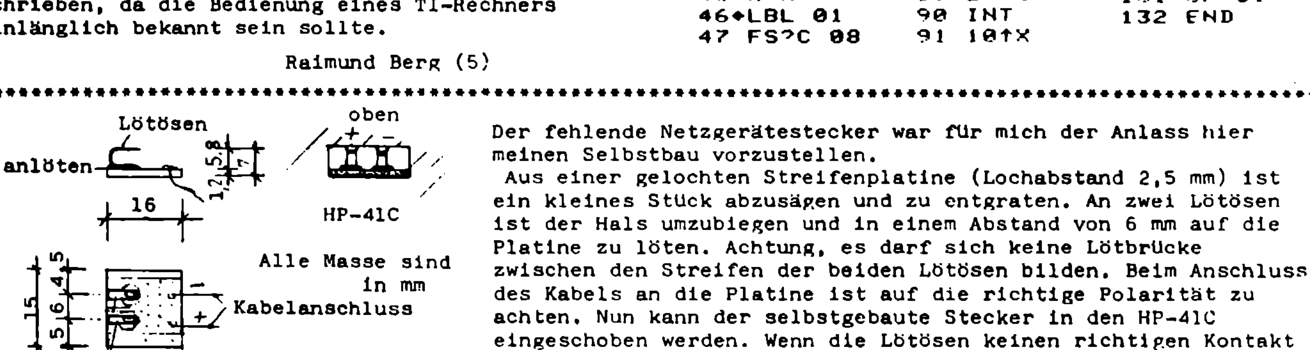
Die Anzahl der unvollständigen Operationen beträgt pro Klammerebene 3.
Die ersten Speicherregister (in der Regel nicht mehr als 18 ≙ 4 Klammerebenen) sollten nicht zur Datenspeicherung benutzt werden.
Die Funktion ᵀINV ist nur in Verbindung mit ᵀYX wirksam.

Die Bedienung des Programms wird hier nicht beschrieben, da die Bedienung eines TI-Rechners hinlänglich bekannt sein sollte.

Raimund Berg (5)

```
01♦LBL "+"       48 1/X           92 FC?C IND Y
02 XEQ 03        49 FS?C 07       93 CLX
03 SF 05         50 *             94 ST+ Z
04 RTN           51 RTN           95 RDN
05♦LBL "-"       52♦LBL "CLR"     96 ISG X
06 XEQ "+"       53 .01           97 GTO 04
07 SF 06         54♦LBL 06        98 RDN
08 RTN           55 CF IND X      99 STO IND 01
09♦LBL "*"       56 ISG X         100 SF 00
10 XEQ 00        57 GTO 06        101 RDN
11 XEQ 01        58 CLST          102 RTN
12 SF 07         59 STO 00        103♦LBL ">"
13 RTN           60 STO 01        104 FC? 00
14♦LBL "/"       61 RTN           105 RTN
15 XEQ "*"       62♦LBL "<"       106 XEQ 03
16 SF 08         63 ISG 00        107 10,004
17 RTN           64♦LBL 10        108 RCL IND 01
18♦LBL "YX"      65 RCL 00        109♦LBL 05
19 FS?C 09       66 STO 01        110 RCL Y
20 XEQ 00        67 RDN           111 INT
21 SF 09         68 4             112 10↑X
22 RTN           69 ST* 01        113 X<=Y?
23♦LBL "INV"     70 SQRT          114 SF IND Z
24 SF 10         71 ST- 01        115 X>Y?
25 RTN           72 RDN           116 CLX
26♦LBL "="       73 STO IND 01    117 -
27 FS? 00        74 RDN           118 DSE Y
28 XEQ ">"       75 ISG 01        119 GTO 05
29 FS? 00        76♦LBL 10        120 RCL Z
30 GTO "="       77 STO IND 01    121 DSE 01
31♦LBL 03        78 RDN           122 RCL IND 01
32 XEQ 00        79 ISG 01        123 DSE 01
33 XEQ 01        80♦LBL 10        124 RCL IND 01
34♦LBL 02        81 STO IND 01    125 DSE 01
35 FS?C 06       82 RCL Z         126 RCL IND 01
36 CHS           83 ISG 01        127 DSE 01
37 FS?C 05       84♦LBL 10        128 R↑
38 +             85 0             129 DSE 00
39 RTN           86 ENTER↑        130 RTN
40♦LBL 00        87 5,01          131 CF 00
41 FS?C 10       88♦LBL 04        132 END
42 1/X           89 ENTER↑
43 FS?C 09       90 INT
44 Y↑X           91 10↑X
45 RTN
46♦LBL 01
47 FS?C 08
```

***



Der fehlende Netzgerätestecker war für mich der Anlass hier meinen Selbstbau vorzustellen.
Aus einer gelochten Streifenplatine (Lochabstand 2,5 mm) ist ein kleines Stück abzusägen und zu entgraten. An zwei Lötösen ist der Hals umzubiegen und in einem Abstand von 6 mm auf die Platine zu löten. Achtung, es darf sich keine Lötbrücke zwischen den Streifen der beiden Lötösen bilden. Beim Anschluss des Kabels an die Platine ist auf die richtige Polarität zu achten. Nun kann der selbstgebaute Stecker in den HP-41C eingeschoben werden. Wenn die Lötösen keinen richtigen Kontakt herstellen, können sie durch aufbiegen gerichtet werden.

Walter Pieperhoff (205)

***

```
01♦LBL "U"
02 CLST
03 LASTX
04 X<> Z
05♦LBL 01
06 RDN
07 1
08 ST+ T
09 RDN
10 STOP
11 STO IND T
12 DSE Z
13 GTO 01
14 END
```

Nebenstehend ein Unterprogramm zur Eingabe von Ziffern. Das Programm kann von mehreren Hauptprogrammen aus aufgerufen werden, wenn nacheinander viele Parameter benötigt werden. Vom Hauptprogramm wird es mit XEQ U aufgerufen. Es werden nacheinander 6 Werte angefordert und dann in die Speicherregister 1 bis 7 abgelegt.

Walter Pieperhoff (205)

***

Danke !

Im Namen aller Mitglieder möchte ich Fred Huber (186) vielmals für die geleistete Arbeit danken ! Er hat für dieses Info sieben Blätter (14 Seiten) abdruckreif übertragen und mir zugesandt ! Also: Vielen Dank, Fred !!

Oliver (1)

***

Geplant ist ein Lösungsbuch zum Thema Hochbau (Bauing.wesen) ohne Statik. Wer dazu Programme zu bieten hat, sende diese bitte an: Jürgen Hüneke (419), Gerhard-Rohlfs-Str. 79, Postfach 7oo 443, 282o Bremen 7o, Tel. o421-65o125. Es liegen bereits vor: Berechnung des umbauten Raumes nach DIN 277, Berechnung des Wärmebedarfs für Gebäude nach DIN 47o1, Auslegung von DIN-Radiatoren, Erstellen eines Terminplanes nach Eingabe von: Gewerksnamen, Gewerksdauer in Arbeitstagen, Vorläufer, Nachfolger, und Abstand Beginn-Beginn zum Vorläufer. Ausgedruckt wird die Gesamtdauer in Wochentagen und wenn erwünscht die einzelnen Ausführungszeiten. Das Lösungsbuch Statik wird von Walter Pieperhoff (2o5) erstellt.

Walter Kropf (56)

Nachstehend einige Betrachtungen zum "Programmierbaren Byte Jumper" und dessen Anwendung zur Verschlüsselung von Programmen.
(Siehe prisma 41/81 und 113/81)

Die Idee, Programmbytes in Textzeilen zu verstecken ist bestimmt eine effektive Methode zur Verschlüsselung von Programmen, welche dadurch kaum mehr lesbar sind. Leider hat die Methode einige Nachteile:

1.) Die Programme werden durch den PBJ sehr langsam, da bei jedem GTO "" oder XEQ "" der ganze Katalog durchgesucht werden muß.

2.) Jede Textzeile erfordert einen Mehraufwand von 5 Bytes. (SF 25, GTO "", Fn)

3.) Wenn das Programm im SST abgefahren wird, sind alle Befehle im Klartext zu lesen! FS? 51, STO c, wie im Programm Seite 113 angewendet hilft auch nicht, wenn nach den ersten Zeilen mit SST begonnen wird. Es müßte am Beginn jeder Textzeile FC? 51 eingegeben werden, doch ist dies wieder ein Mehraufwand von 2 Bytes je Textzeile.

4.) Beim Programmlauf im TRACE-Modus werden alle Programmschritte im Klartext ausgedruckt, damit ist die ganze Verschlüsselung umsonst.
Ich habe dagegen noch kein Mittel gefunden, obwohl der Rechner den TRACE-Modus irgendwie erkennen muß, bei PRIVATE ist die Ausgabe im TRACE-Modus gesperrt. Wer weiß mehr?

Für diejenigen, die auf diesem Gebiet weiterarbeiten ein Hinweis: GTO und XEQ Befehle sind in den Textzeilen möglich, doch darf nach dem ersten Programmlauf keine Änderung im Programm mehr erfolgen (vor Programmlauf packen!), da die Sprungweiten wohl richtig berechnet und gespeichert werden, bei einer Editierung aber diese Bytes, wenn sie in Textzeilen stehen, nicht mehr auf 0 gesetzt werden. Am Besten man zeichnet das Programm nach packen vor einem Probelauf auf Magnetkarten auf, dann hat man immer das jungfräuliche Programm mit undefinierten Sprungweiten zur Verfügung.

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

C L U B B Ö R S E

Clubmitglieder können in der Clubbörse kostenlos Gelegenheitsanzeige aufgeben

ALU-Koffer für HP-41c/cv

Ich beabsichtige, eine größere Anzahl (minimal 5) äußerst stabiler Alu-Koffer für das HP-41 System herstellen zu lassen. Geplante Ausstattung:

- umlaufende, massive Alu-Profile mit Dichtungsschnur
- abschließbare Zugschlösser
- Bodennägel und aushängbare Scharniere
- Flächen Alu-Sperrholz-Sandwich, verklebt und vernietet
- Unterteil: Schaumstoff-Inneneinteilung für Rechner, Magnetkartenleser, Barcodeleser, Drucker, Module, Netzteil, Papierrolle, Magnetkartentaschen
- Zwischendeckel
- Oberteil frei verfügbar
- Innenmaße: 400 x 285 x 100 mm
- Preis ca. 200,-- inkl. Mwst. bei 5 Stck., bei höheren Stückzahlen günstiger.

Michael Gehret (266), Silcherstr. 15, 8944 Grönenbach, Tel. 08334/1001

Außerdem verkaufe ich drei SD-RAMs für zusammen 130,--DM.

Ulrich Daver[illegible]fen
Ostenbergstra[illegible]97 / 8
4600 Dortmund 50

Finanzbericht Clubprojekte 1 + 3 , II/1980 und I/1981

1. 2.Halbjahr 1980 (Beginn der Projekte - 31.12.1980) :

Einnahmen :		DM 2470.00	=	100.00 %
Ausgaben :	a) Materialeinkauf	DM 2257.08	=	91.38 %
	b) Verpackung,Porto	DM 91.96	=	3.72 %
	c) Sonstiges	DM 40.75	=	1.65 %
		DM 2389.79	=	96.75 %
Aufwandsentschädigung (= Überschuß):		DM 80.21	=	3.25 %

Zu a) Es wurden 3740 Magnetleerkarten, 94 Kartentaschen und 19 Rollen Thermopapier bestellt, eingekauft und ausgeliefert.

Zu c) Umfaßt: Paketzustell-,Kontogebühren,Porto(Rückfragen, Bestätigungen,Händlerkorrespondenz),Probelieferungen, Bürobedarf,Kopierkosten u.a.

---

2. 1.Halbjahr 1981 (1.1. - 30.6.1981) :

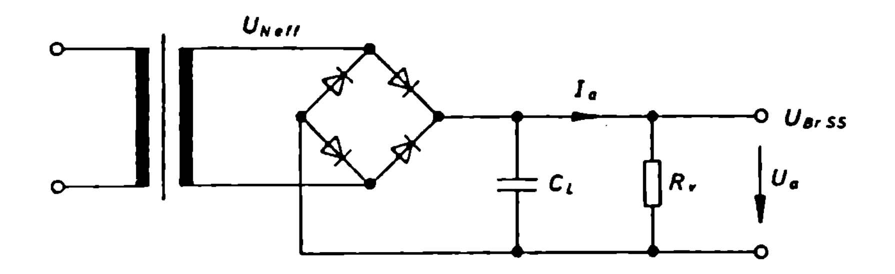
Einnahmen :		DM 11795.85	=	100.00 %
Ausgaben :	a) Materialeinkauf	DM 10541.28	=	89.36 %
	b) Verpackung,Porto	DM 685.76	=	5.81 %
	c) Sonstiges	DM 207.45	=	1.76 %
		DM 11434.49	=	96.93 %
Aufwandsentschädigung (= Überschuß):		DM 361.36	=	3.07 %

Zu a) Es wurden 13480 Magnetleerkarten, 302 Kartentaschen, 193 Einlegeblätter und 1696 Rollen Thermopapier bestellt, eingekauft und ausgeliefert.

Für den Berichtszeitraum ergibt sich ein Überschuß von DM 441.57. Diesen Betrag habe ich als Entlohnung für die von mir verrichteten organisatorischen Arbeiten (von Aufnahme bis Zusendung) einbehalten. Legt man einen Aufwand von 180 Stunden zugrunde (und dies ist noch knapp bemessen), so ergibt sich ein durchschnittlicher Stundenlohn von DM 2.45 .
Die obigen Angaben sind durch Belege,Quittungen,Rechnungen und Kontoauszüge nachweisbar.

Dortmund, 1.August 1981 Ulli Davertzhofen (3)

Berechnung der Trafospannung fu
Brückengleichrichter mit Ladekondensator



$U_{N\,eff}$ = Trafospannung unter Last

f_v = Verlustfaktor vom Trafo

$U_{a\,min}$ = Ausgangsgleichspannung bei I_a

$U_{Br\,ss}$ = Brummspannung an C_L unter Last

R_v = Verbraucherwiderstand

$U_{a\,\infty}$ = Last - Ausgangsgleichspannung (mit ∞ gr. C_L)

U_{a0} = Ausgangsgleichspannung ohne Last

P_N = Trafo - Nennleistung

R_i = Trafo - Innenwiderstand

U_D = Diodenspannung unter Last

I_{DS} = Diodenspitzenstrom

f_N = Netzfrequenz

$I_{N\,eff}$ = Trafo - Ausgangsstrom

2) Rech●ngsgang :

LBL A

$U_{a\infty} = U_{a\,min} + 0{,}67\ U_{B\,ss}$ (Abschätzung)

$P_N = 1{,}2 \cdot I_a\,(U_{a\infty} + 2U_D)$

Kerngröße auswählen

f_v aus der Tabelle entnehmen ⟶ STO 03

R/S

LBL 01

1. Rechnungsgang
$U_{a\,min} = U_{N\,eff}$

$$R_i = \frac{U_{N\,eff}^2}{P_N}\,(f_v - 1)$$

$$R_v = \frac{U_{a\infty}}{I_a}$$

$$U_{a\infty} = \left(\sqrt{2} \cdot U_{N\,eff} \cdot f_v - 2U_D\right)\left(1 - \sqrt{\frac{R_i}{2R_v}}\right)$$

$U_{a\infty}$ (Schätzwert) = $U_{a\infty}$ (2. Rechnungsgang) = Differenz

= Differenz + $U_{N\,eff}$ (bis die Differenz kleingeworden ist)

LBL 02

$$C_L = \frac{I_a}{100 \cdot U_{Br\,ss}} \cdot \left(1 - \sqrt[4]{\frac{R_i}{2R_v}}\right)$$

$$U_{Br\,ss} = \frac{I_a}{100 \cdot C_L} \cdot \left(1 - \sqrt[4]{\frac{R_i}{2R_v}}\right)$$ für C_L Normwert eingeben

$$I_{N\,eff} = \frac{P_N}{U_{N\,eff}}\ ; \quad U_{a0} = \sqrt{2} \cdot U_{N\,eff} \cdot f_v - 2U_D$$

$$I_{DS} = \frac{U_{a0}}{\sqrt{2R_i \cdot R_v}}$$

H. Henze
PRGM-HP41C
GOETTINGEN 10.80

Berechnung der Trafospannung für Brückengleichrichter

```
01•LBL "GL"
02•LBL A
03 RCL 00
04 ,67
05 RCL 01
06 *
07 +
08 STO 05
09 2
10 +
11 RCL 02
12 RCL 04
13 *
14 *
15 STO 06
16 "PN="
17 ARCL X
18 PROMPT
19 RCL 00
20 STO 07

21•LBL 01
22 RCL 07
23 X↑2
24 RCL 06
25 /
26 RCL 03
27 1
28 -
29 *
30 STO 08
31 2
32 SQRT
33 RCL 07
34 *
35 RCL 03
36 *
37 2
38 -
39 RCL 05
40 RCL 02
41 /
42 STO 09
43 2
44 *
45 RCL 08
46 X<>Y
47 /
48 STO 10
49 SQRT
50 1
51 X<>Y
52 -
53 *
54 STO 15
55 RCL 05
56 X<>Y
57 -
58 ST+ 07
59 ,1
60 X>Y?
61 GTO 02
62 GTO 01

63•LBL 02
64 RCL 02
65 100
66 RCL 01
67 *
68 /
69 RCL 10
70 4
71 1/X
72 Y↑X
73 1
74 X<>Y
75 -
76 STO 11
77 *
78 "CL="
79 ARCL X
80 PROMPT
81 "NORMW=?"

82 PROMPT
83 STO 12
84 RCL 02
85 100
86 RCL 12
87 *
88 /
89 RCL 11
90 *
91 STO 13
92 2
93 SQRT
94 RCL 07
95 *
96 RCL 03
97 *
98 2
99 -
100 STO 14
101 2
102 RCL 08
103 *
104 RCL 09
105 *
106 SQRT
107 /
108 "ID="
109 ARCL X
110 PROMPT
111 RCL 07
112 "UN="
113 ARCL X
114 PROMPT
115 RCL 15
116 2
117 +
118 RCL 02
119 *
120 1,2
121 *
122 STO 06
123 RCL 07
124 /
125 "IN="
126 ARCL X
127 PROMPT
128 RCL 06
129 "PN="
130 ARCL X
131 PROMPT
132 END
```

Eingeben:

- 00 U_a min
- 01 $U_{Br\ ss}$
- 02 I_a
- 03 f_v
- 04 Formfaktor [1,2]

05| U_a =

06| P_n

07| U_{Neff}

08| Ri

09| R_v

10| $\frac{Ri}{2 \cdot R_v}$

11| $1 - \sqrt[4]{\frac{Ri}{2 \cdot R_v}}$

12| C_L

13| $U_{Br\ ss}$

14| $U_{a\ o}$

15| U_a ~ gerechn.

Programmbeschreibung GL

Dieser Rechnungsgang ist sehr gut beschrieben in dem
Buch : Halbleiter - Schaltungstechnik 5. Auflage
von U. Tietze u. Ch. Schenk
Springer Verlag Berlin 1980 Seite 366

Henze
RGM-HP41C
ETTINGEN 3/81

Trafo – Berechnung

```
01◆LBL "TRF"
02 "UP?"
03 PROMPT
04 STO 00
05 "US1?"
06 PROMPT
07 STO 01
08 "US2?"
09 PROMPT
10 STO 02
11 "US3?"
12 PROMPT
13 STO 03
14 "IS1?"
15 PROMPT
16 STO 04
17 "IS2?"
18 PROMPT
19 STO 05
20 "IS3?"
21 PROMPT
22 STO 06
23 "V?"
24 PROMPT
25 STO 07
26 "S?"
27 PROMPT
28 STO 08
29 "QFE?"
30 PROMPT
31 STO 09
32◆LBL A
33 26,5      ← 1,4T
34 RCL 09
35 /
36 STO 10
37 RCL 00
38 *
39 STO 11
40 "WP="
41 XEQ 00
42 RCL 01
43 XEQ 01
44 STO 12
45 "WS1="
46 XEQ 00
47 RCL 02
48 XEQ 01
49 STO 13
50 "WS2="
51 XEQ 00
52 RCL 03
53 XEQ 01
54 STO 14
55 "WS3="
56 XEQ 00
57 RCL 01
58 RCL 04
59 *
60 RCL 02
61 RCL 05
62 *
63 +
64 RCL 03
65 RCL 06
66 *
67 +
68 RCL 07
69 *
70 "P="
71 XEQ 00
72 RCL 00
73 /
74 "IP="
75 XEQ 00
76 XEQ 02
77 STO 15
78 RCL 04
79 XEQ 02
80 STO 16
81 RCL 05
82 XEQ 02
83 STO 17
84 RCL 06
85 XEQ 02
86 STO 18
87 RCL 14
88 *
89 RCL 17
90 RCL 13
91 *
92 +
93 RCL 16
94 RCL 12
95 *
96 +
97 RCL 15
98 RCL 11
99 *
100 +
```

Schnittband Kern 1,7T – 1,83T

Dynamoblech IV 1,2T

```
101  .5
102  /
103  "QF="
104  XEQ 00
105  RCL 15
106  XEQ 03
107  "DP="
108  XEQ 00
109  RCL 16
110  XEQ 03
111  "DS1="
112  XEQ 00
113  RCL 17
114  XEQ 03
115  "DS2="
116  XEQ 00
117  RCL 18
118  XEQ 03
119  "DS3="
120  XEQ 00
121♦LBL 00
122  ARCL X
123  AVIEW
124  STOP
125  RTN
126♦LBL 01
127  RCL 10
128  *
129  RCL 07
130  *
131  RTN
132♦LBL 02
133  RCL 08
134  /
135  RTN
136♦LBL 03
137  4
138  *
139  PI
140  /
141  SQRT
142  RTN
143  END
```

```
R00= "UP"
R01= "US1"
R02= "US2"
R03= "US3"
R04= "IS1"
R05= "IS2"
R06= "IS3"
R07= "VERL."
R08= "S" A/mm
R09= "QFE" cm²
R10= "WDG/V"
R11= "WP"
R12= "WS1"
R13= "WS2"
R14= "WS3"
R15= "QP" mm²
R16= "QS1" "
R17= "QS2" "
R18= "QS3" "
```

$$\frac{Wdg}{V} = \frac{26,5}{QFE\ [cm^2]}$$

(für 1,7T, 50Hz)

$$\frac{Wdg}{V} = \frac{40}{QFE\ [cm^2]}$$

(für 1,2T, 50Hz)

Drahtquerschnitt

$$q = \frac{J}{S}$$

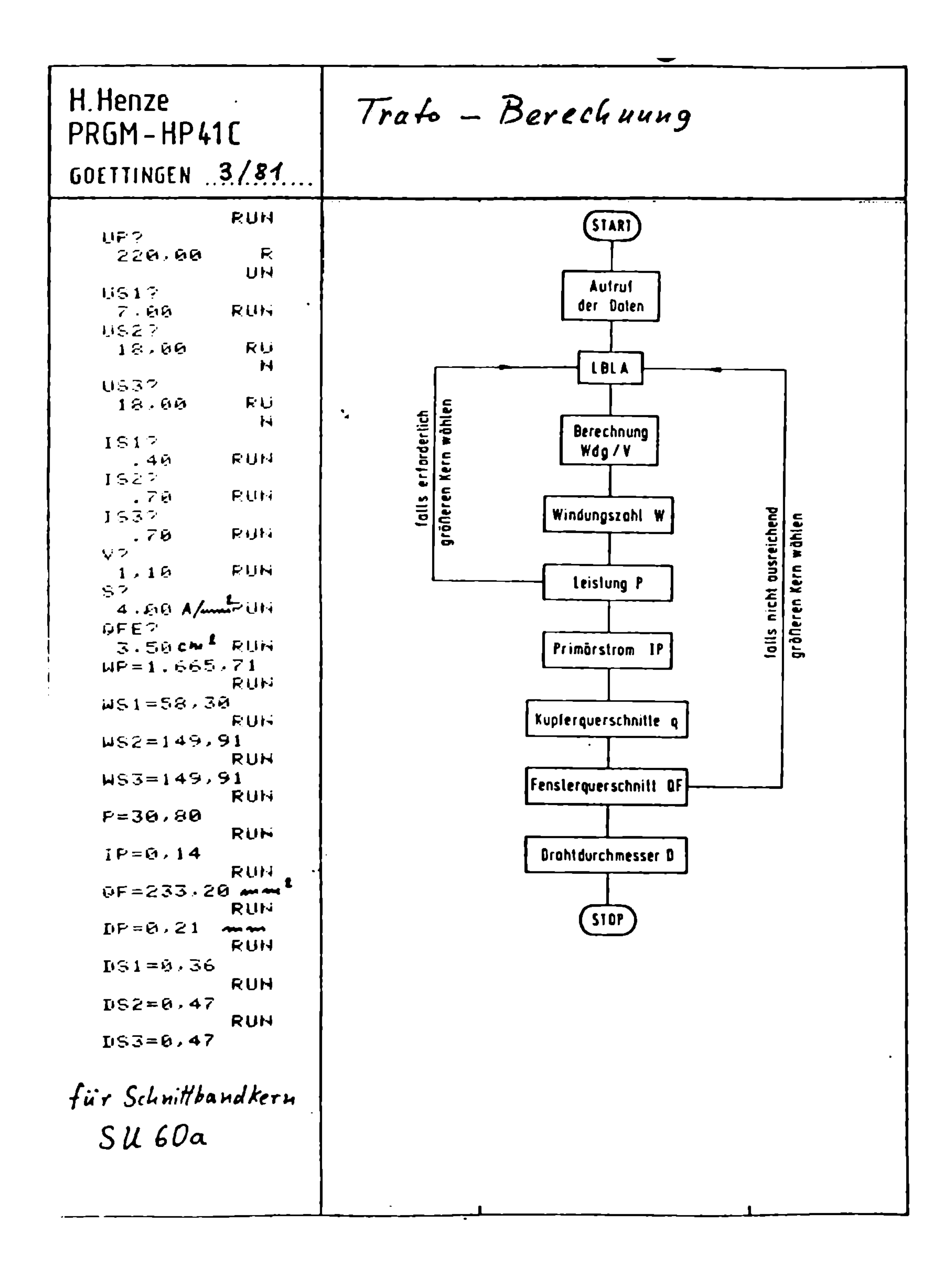
Fensterquerschnitt

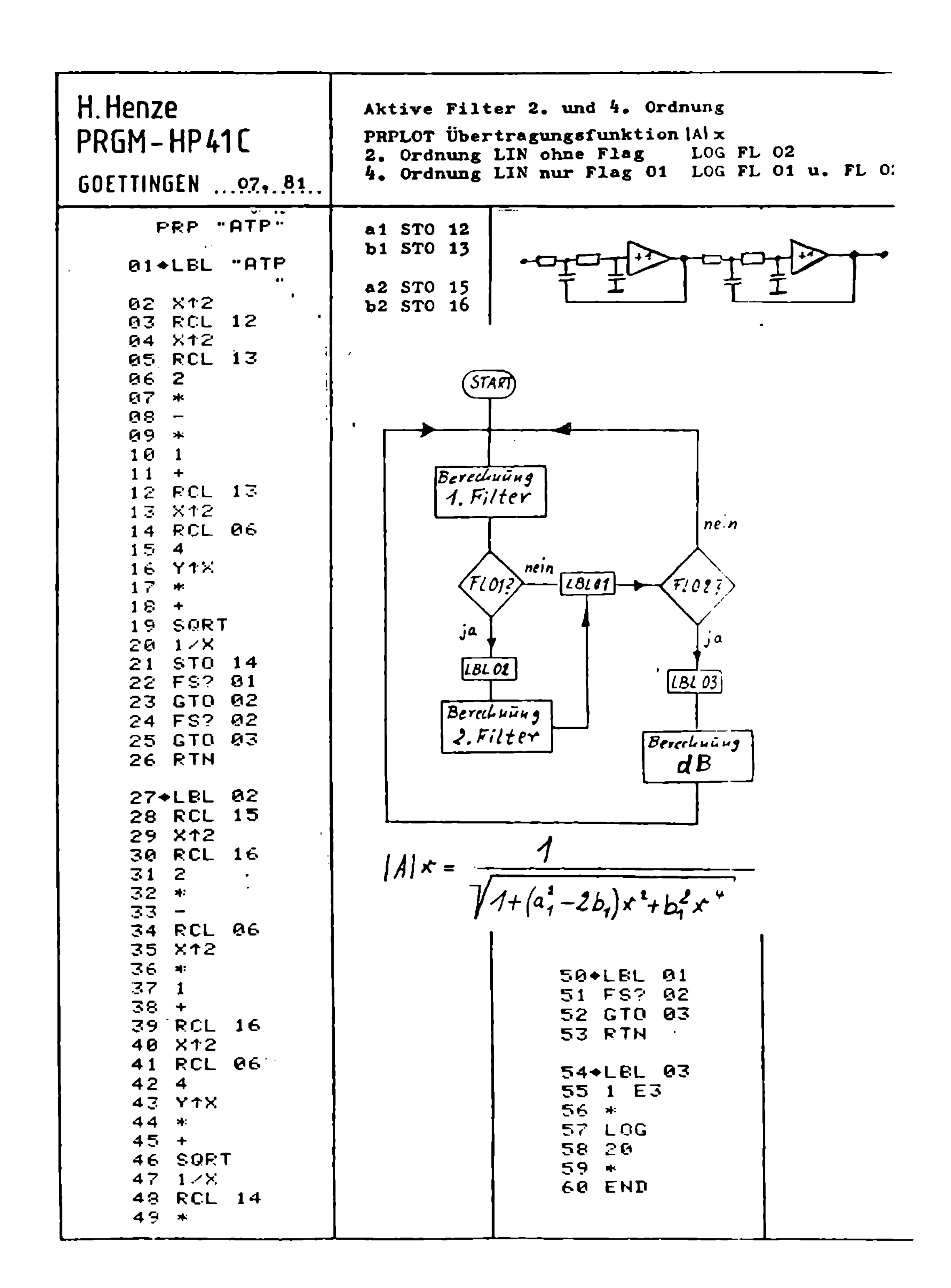
$$QF = WP \times qp + WS1 \times qs1 + WS2 \times qs2 + WS3 \times qs3$$

Drahtdurchmesser

$$D = \sqrt{\frac{4 \times q}{\pi}}$$

UP
US1
US2
US3


H.Henze
PRGM-HP41C
GOETTINGEN 3/81
Trafo – Berechnung
RUN
UP?
220,00 RUN
US1?
7.00 RUN
US2?
18,00 RUN
US3?
18.00 RUN
IS1?
.40 RUN
IS2?
.70 RUN
IS3?
.70 RUN
V?
1,10 RUN
S?
4.00 A/mm² RUN
QFE?
3.50 cm² RUN
WP=1.665,71
RUN
WS1=58,30
RUN
WS2=149,91
RUN
WS3=149,91
RUN
P=30,80
RUN
IP=0,14
RUN
QF=233.20 mm²
RUN
DP=0,21 mm
RUN
DS1=0,36
RUN
DS2=0,47
RUN
DS3=0,47
für Schnittbandkern
SU 60a
START
Aufruf der Daten
LBL A
Berechnung Wdg/V
Windungszahl W
Leistung P
Primärstrom IP
Kupferquerschnitte q
Fensterquerschnitt QF
Drahtdurchmesser D
STOP
falls erforderlich größeren Kern wählen
falls nicht ausreichend größeren Kern wählen



H. Henze
PRGM-HP41C
GOETTINGEN ...07. 81..

Aktive Filter 2. und 4. Ordnung

PRPLOT Übertragungsfunktion |A| x

2. Ordnung	LIN ohne Flag	LOG FL 02
4. Ordnung	LIN nur Flag 01	LOG FL 01 u. FL 0

PRP "ATP"

```
01♦LBL "ATP"
02 X↑2
03 RCL 12
04 X↑2
05 RCL 13
06 2
07 *
08 -
09 *
10 1
11 +
12 RCL 13
13 X↑2
14 RCL 06
15 4
16 Y↑X
17 *
18 +
19 SQRT
20 1/X
21 STO 14
22 FS? 01
23 GTO 02
24 FS? 02
25 GTO 03
26 RTN

27♦LBL 02
28 RCL 15
29 X↑2
30 RCL 16
31 2
32 *
33 -
34 RCL 06
35 X↑2
36 *
37 1
38 +
39 RCL 16
40 X↑2
41 RCL 06
42 4
43 Y↑X
44 *
45 +
46 SQRT
47 1/X
48 RCL 14
49 *
```

```
a1 STO 12
b1 STO 13

a2 STO 15
b2 STO 16
```


$$|A|x = \frac{1}{\sqrt{1+(a_1^2-2b_1)x^2+b_1^2x^4}}$$

```
50♦LBL 01
51 FS? 02
52 GTO 03
53 RTN

54♦LBL 03
55 1 E3
56 *
57 LOG
58 20
59 *
60 END
```

H.Henze
PRGM-HP41C
GOETTINGEN ..07..81...

TSCHEBYSCHEFF FILTER 4. Ordnung
mit 0,5 dB Welligkeit
a1 = 2,6282 b1 = 3,4341 (1. Filter)
a2 = 0,3648 b2 = 1,1509 (2. Filter)

```
          2.63 STO 12
          3.43 STO 13
           .36 STO 15
          1.15 STO 16
         XROM "PPPLOT"

NAME ?
ATP               RUN
Y MIN ?
         20.00    RUN
Y MAX ?
         80.00    RUN
AXIS ?
         20.00    RUN
X MIN ?
          0.00    RUN
X MAX ?
          5.00    RUN
X INC ?
           .20    RUN

   PLOT OF ATP
  X (UNITS= 1.) ↓
  Y (UNITS= 1.) →
    20.0       80.0
    20.0
    |--------------|  dB
0.00:        *
0.20:        *
0.40:        *
0.60:        *
0.80:        *
1.00:        *
1.20:      *
1.40:    *
1.60:   *
1.80:  *
2.00: *
2.20:*
2.40 *
2.60 *
2.80 *
3.00 *
3.20 *
3.40 *
3.60 *
3.80 *
4.00 *
4.20 *
4.40 *
4.60 *
4.80 *
5.00 *
```

1.+2. Filter

```
           .36 STO 12
          1.15 STO 13
         XROM "PPPLOT"

NAME ?
ATP               RUN
Y MIN ?
         20.00    RUN
Y MAX ?
         80.00    RUN
AXIS ?
         20.00    RUN
X MIN ?
          0.00    RUN
X MAX ?
          5.00    RUN
X INC ?
           .20    RUN

   PLOT OF ATP
  X (UNITS= 1.) ↓
  Y (UNITS= 1.) →
    20.0       80.0
    20.0
    |--------------|  dB
0.00:          *
0.20:          *
0.40:          *
0.60:           *
0.80:           *
1.00:           *
1.20:          *
1.40:         *
1.60:        *
1.80:        *
2.00:       *
2.20:       *
2.40:      *
2.60:      *
2.80:      *
3.00:     *
3.20:     *
3.40:     *
3.60:    *
3.80:    *
4.00:    *
4.20:   *
4.40:   *
4.60:   *
4.80:   *
5.00:   *
```

2. Filter

```
          2.63 STO 12
          3.43 STO 13
         XROM "PPPLOT"

NAME ?
ATP               RUN
Y MIN ?
         20.00    RUN
Y MAX ?
         80.00    RUN
AXIS ?
         20.00    RUN
X MIN ?
          0.00    RUN
X MAX ?
          5.00    RUN
X INC ?
           .20    RUN

   PLOT OF ATP
  X (UNITS= 1.) ↓
  Y (UNITS= 1.) →
    20.0       80.0
    20.0
    |--------------|  dB
0.00:          *
0.20:          *
0.40:          *
0.60:         *
0.80:         *
1.00:        *
1.20:       *
1.40:       *
1.60:      *
1.80:      *
2.00:     *
2.20:     *
2.40:     *
2.60:    *
2.80:    *
3.00:    *
3.20:   *
3.40:   *
3.60:  *
3.80:  *
4.00: *
4.20: *
4.40: *
4.60:*
4.80:*
5.00 *
```

1. Filter

H.Henze
PRGM-HP41C
GOETTINGEN 08. 81

Aktive Filter 2. Ordnung

PRPLOT Phase vom Tiefpass

```
          SF 12
   PRP "PTF"

 01♦LBL "PTF"
 02 RCL 12
 03 *
 04 1
 05 RCL 13
 06 RCL 06
 07 X↑2
 08 *
 09 -
 10 /
 11 ATAN
 12 CHS
 13 X>0?
 14 GTO 01
 15 RTN
 16♦LBL 01
 17 -180
 18 +
 19 END

 2,63 STO 12
 3,43 STO 13
```

$$\varphi = -\arctan \frac{a_1 x}{1 - b_1 x^2}$$

```
          XROM "PRPLOT"
NAME ?
PTF                 RUN
Y MIN ?
          -180,00   RUN
Y MAX ?
             0,00   RUN
AXIS ?
             0,00   RUN
X MIN ?
             0,00   RUN
X MAX ?
             2,00   RUN
X INC ?
              ,05   RUN

      PLOT OF PTF
     X (UNITS= 1,) ↓
     Y (UNITS= 1,) →
     -180,          0
                    0.
        |-----------|
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
0.45
0.50
0.55
0.60
0.65
0.70
0.75
0.80
0.85
0.90
0.95
1.00
1.05
1.10
1.15
1.20
1.25
1.30
1.35
1.40
1.45
1.50
```

Walter Kropf, Schönowitzstraße 14, A-8700 Leoben

Das Programm berechnet für ein bestimmtes Jahr das Datum von Vollmond bzw. Neumond. Das Programm erhebt keinen Anspruch auf astronomische Genauigkeit. Abweichungen des berechneten Datums vom tatsächlichen Ereignis können +– 1 Tag betragen.

Das Programm benötigt als Unterprogramme die Programme "Kalenderrechnungen" KA101 und "Bewegliche Feste" KA102.

STATUS: SIZE 012
USER ON

Tastenzuordnung: LBL A START
LBL B Vollmond
LBL C Neumond

Programmspeicher:
165 Bytes, 24Reg.
Unterprogr.KA101 64Reg.
" KA102 53Reg.

Speicherliste:
R00) vom Programm KA101
R06) belegt
R07 ?Vollmond im Jahr
R08) vom Programm KA102
R11) belegt

Flags:
01 Neumond
02 Steuerung des UP KA102
03) Steuerung des UP KA101
04)

Das Datum wird in der Form MM,TTJJJJ angezeigt.

Programmablauf:
START mit Taste A, nach Aufforderung durch den Rechner das Jahr eingeben, der Rechner hält mit der Anzeige der goldenen Zahl an, wenn kein Drucker angeschlossen ist, mit R/S fortsetzen. In der Anzeige erscheint die Tastenzuordnung für die Programmfortsetzung. Zur Berechnung der Neumond-Daten Taste C, zur Berechnung der Vollmond-Daten Taste B drücken. Auf die Abfrage '?VM' ('?NM') ist einzugeben, der wievielte Vollmond (Neumond) im laufenden Jahr berechnet werden soll. Das Ergebnis wird angezeigt, bzw. gedruckt. Mit R/S kann der nächste Termin berechnet werden. Wenn die Unterprogramme nicht schon geladen sind, sind sie nach Aufforderung durch den Rechner einzulesen.

Rechnerkonfiguration:
2 Memorymodule
(Drucker)
(Kartenleser)

```
01•LBL "MOND"
02•LBL A          Programmstart
03 SF 02
04•LBL 03
05 SF 25
06 XEQ "BEWFEST"  Aufr.UP
07 FC?C 25
08 GTO 08
09 30,55
10 ST- 02         1.VM im Jahr
[illegible]       Tastenzuordng.
[illegible]
[illegible]
[illegible]
[illegible]
16 PROMPT
17 GTO 01
18•LBL B
19 CF 01          CF01 für VM
20 "?VM"
21 PROMPT
22•LBL 01
23 STO 07
24 13
25 RCL 07
26 1
27 -
28 X>Y?
29 GTO 02         Ende des Jahres
30 29,517
31 *
32 RCL 02
33 +
34 -15
35 X<>Y
36 FS? 01
37 +
38 FIX 0
[illegible]
[illegible]       Tageszahl [illegible]
[illegible]
[illegible]
[illegible]
44 XEQ "KAL"      Datum berechn.
45 ASTO Y
46 FC? 01
47 "VM"
48 FS? 01
49 "NM"
50 FS? 55
51 "⊢ "
52 ARCL Y
53 ARCL X
54 AVIEW          Datum anzeigen
55 1
56 ST+ 07         Ereignis +1
57 RCL 07
58 GTO 01         nächstes Ereignis
59•LBL 08
60 CF 21          UP einlesen
61 "RPRG. BEWFEST"
62 AVIEW
63 RSUB
64 GTO 03
65•LBL 02         Programmende
66 BEEP
67 FIX 2
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
LBL "MONL
END               165 BYTES
```

Walter Kropf, S[illegible]itzstr.14 A-8700 Leoben Regelkalender KA108

Das Programm berechnet aus den Eingaben: längste Periode, kürzeste Periode, letztes Regeldatum die fruchtbaren Tage und den Zeitpunkt der nächsten Regel nach der Methode Knaus-Ogino. Für die Anwendung des Programms ist auf die einschlägige Literatur zurückzugreifen. Wenn ein Drucker angeschlossen ist, wird ein Kalender für die nächsten 40 Tage mit Kennzeichnung der fruchtbaren Tage und des Eintritts der nächsten Regel ausgedruckt. Der Druckstreifen enthält auch ein Diagramm zur Eintragung nach der Zyklotest-Methode.

Die Anzahl der Tage zur Berechnung der fruchtbaren Tage ist in Programmzeile 91 und 98 gespeichert. Es lassen sich hier leicht Änderungen vornehmen.

Das Programm verwendet als Unterprogramm das Programm "Kalenderrechnungen" KA101.

STATUS: SIZE 015
USER ON

Tastenzuordnung: LBL A: neues Regeldatum eingeben
LBL B: fruchtbare Tage
LBL C: nächste Regel
LBL E: Regelkalender drucken

Programmspeicher:
454 Bytes, 65Reg.

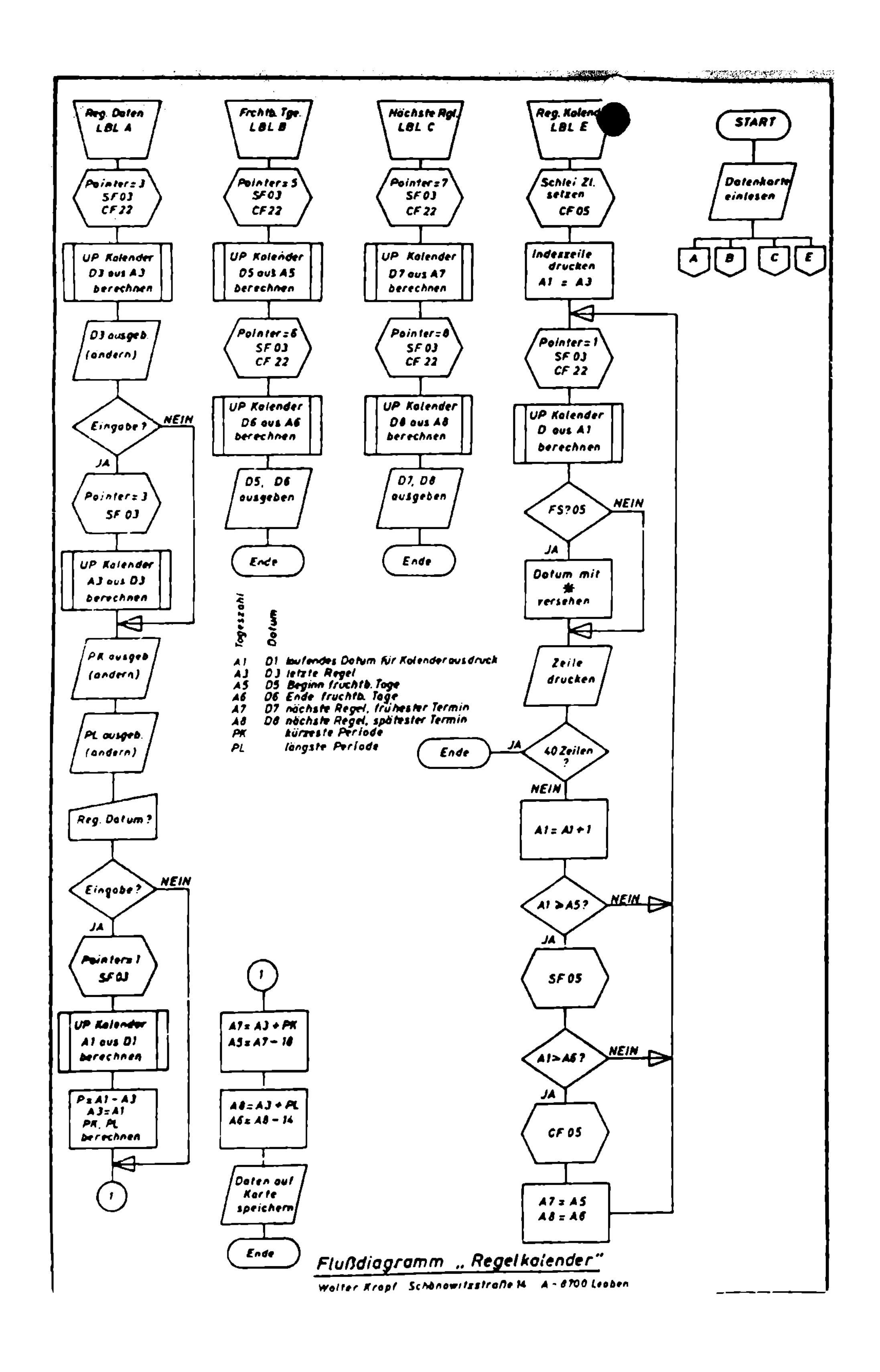
Speicherliste:
R00–R06 Programm KA101
R07 Zwischenspeicher
R08 letzte Regel A3
R09 kürzeste Periode PK
R10 längste Periode PL
R11 Beginn der fruchtb. Tg. A5
R12 Ende der fruchtb. Tg. A6
R13 Nächste Regel frühest A7
R14 Nächste Regel spätest A8

Das Datum ist in der Form MM,TTJJJJ einzugeben. Falls das Programm KA101 nicht schon im Speicher steht, ist es nach Aufforderung durch den Rechner einzulesen.

Programmablauf:
XEQ 'REKA', nach Aufforderung 'RDAT' Karte mit Datenspur einschieben. Es wird die Tastenzuordnung angezeigt.
Eingabe eines neuen Regeldatums: A, angezeigte Werte können durch Eingabe korrigiert werden, mit R/S fortsetzen bis zur Aufforderung 'REG.-DAT?'. Neues Regeldatum eingeben, R/S, nach Aufforderung 'WDAT' Datenkarte zum Festhalten der neuen Werte einschieben.
Anzeige der fruchtbaren Tage: B
Anzeige der zu erwartenden nächsten Regel: C
Ausdruck eines Regelkalenders: E
Für die Anlage einer neuen Datenkarte ist wie unter "Eingabe eines neuen Regeldatums" zu verfahren. Es werden im allgemeinen zufällig im Speicher stehende Werte angezeigt. Es müssen die richtigen Werte für 'L.RG.' (letztes Regeldatum) eingegeben werden. Für 'MIN.P.' und 'MAX.P.' ist, falls noch keine Erfahrungswerte vorliegen, 28 einzugeben. Diese Werte werden dann bei jedem neuen Regeldatum automatisch korrigiert.
Am Ende jeder Berechnung kann mit R/S die Tastenzuordnung in die Anzeige gerufen werden.
Für die Anwendung der Programmteile B und C braucht nur die Spur 1 und 2 des Programms im Rechner zu stehen. Wenn kein Drucker verwendet wird, braucht das Programm nur bis Zeile 125 eingegeben werden.

Rechnerkonfiguration:
2 [illegible]
[illegible]
[illegible]

Achtung ! Die Prgr. KA 105 und KA 106 benötigen die im Kalender-[illegible]ungsbuch enthaltenen Unterprogramme ! Sie werden hier als Ergänzung zum Lösungsbuch verstanden.



Flußdiagramm „Regelkalender"

Walter Kropf Schönowitzstraße 14 A-8700 Leoben

Programm „Re[illegible]kalender"

Walter Kropf Schön[illegible]straße 14 A-8700 Leoben

```
01•LBL "REKA"
02 "RDAT"
03 AVIEW
04 RDTA
05•LBL 05
06 "RB FT NR  RK"
07 PROMPT
08•LBL C
09 "N. REG."
10 14
11 GTO 01
12•LBL B
13 "F. TAGE"
14 12
15•LBL 01
16 FC? 55
17 CF 21
18 AVIEW
19 CF 22
20 XEQ 09
21 ASTO 07
22 RCL 06
23 STO 02
24 RCL 00
25 1
26 -
27 XEQ 09
28 FIX 2
29 ARCL 06
30 "⊦:"
31 ARCL 07
32 ARCL 02
33 AVIEW
34 GTO 05
35•LBL 09
36 SF 03
37 SF 25
39 XEQ "KAL"
39 SF 21
40 FS?C 25
41 RTN
42 CF 21
43 "RPRGR. KAL"
44 AVIEW
45 RSUB
46 GTO 09
47•LBL A
48 CF 22
49 8
50 XEQ 09
51 "L.RG."
52 FIX 6
53 6
54 XEQ 00
55 8
56 FS? 22
57 XEQ 09
58 FIX 0
59 "MIN.P.="
60 9
61 XEQ 00
62 "MAX.P.="
63 10
64 XEQ 00
65 FIX 6
66 6
67 STO 07
68 "REG.-DAT?"
69 XEQ 04
70 FC? 22
71 GTO 03
72 1
73 XEQ 09
74 RCL 01
75 ENTER↑
76 X<> 08
77 -
78 RCL 09
79 X<>Y
80 X<Y?
81 X<> 09
82 RCL 10
83 X<>Y
84 X>Y?
85 X<> 10
86•LBL 03
87 RCL 08
88 RCL 09
89 +
90 STO 13
91 10
92 -
93 STO 11
94 RCL 08
95 RCL 10
96 +
97 STO 14
98 14
99 -
100 STO 12
101 CF 21
102 FIX 0
103 "WDAT. "
104 ARCL 09
105 "⊦:"
106 ARCL 10
107 ,014
108 FIX 2
109 AVIEW
110 WDTAX
111 GTO 05
112•LBL 00
113 STO 07
114 ARCL IND 07
115•LBL 04
116 CF 22
117 AVIEW
118 FS? 55
119 STOP
120 FC? 22
121 RTN
122 FS? 55
123 PRX
124 STO IND 07
125 RTN
126•LBL E
127 SF 21
128 SF 06
129 11,014
130 STO 02
131 1,04
132 STO 07
133 12
134 SKPCHR
135 "-4 36,9 +4"
136 ACA
137 PRBUF
138 SF 01
139 CF 22
140 CF 05
141 RCL 08
142 STO 01
143 SF 28
144•LBL 08
145 1
146 XEQ 09
147 ASTO X
148 RCL 07
149 CLA
150 FIX 0
151 10
152 X>Y?
153 " "
154 ARCL Y
155 ARCL Z
156 FIX 2
157 FS? 01
158 FIX 6
159 ARCL 06
160 ACA
161 FC?C 01
162 XEQ 07
163 PRBUF
164 1
165 RCL IND 02
166 FS? 05
167 +
168 1
169 RCL 01
170 +
171 STO 01
172 X<>Y
173 X<=Y?
174 XEQ 06
175 X<>Y
176 ISG 07
177 GTO 08
178 CF 05
179 CF 28
180 ADV
181 ADV
182 ADV
183 ADV
184 ADV
195 BEEP
186 RTN
197•LBL 06
188 FC?C 06
189 RTN
190 ISG 02
191 SF 06
192 FS?C 05
193 RTN
194 SF 05
195 RTN
196•LBL 07
197 42
198 FS? 05
199 ACCHR
200 2
201 FS? 05
202 1
203 SKPCHR
204 46
205 ACCHR
206 ACCHR
207 ACCHR
208 ACCHR
209 124
210 ACCHR
211 X<>Y
212 ACCHR
213 ACCHR
214 ACCHR
215 ACCHR
216 END
LBL'REKA
END          454
```

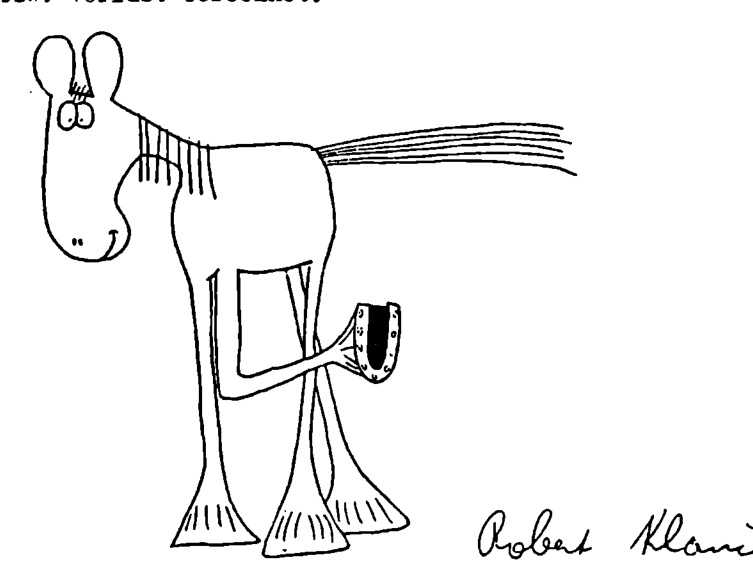
System-Konfiguration: 2 Memory Module, Printer.

DERBY

Dieses Programm simuliert ein Pferderennen über 1 Meile für 3 jährige Vollblute. Der Spieler kann bis zu zehn Wetten auf die 8 Pferde setzen.
Es gibt 3 verschiedene Wettarten (Positionen):

1) auf Sieg (1)
2) auf ersten und zweiten Platz (2)
3) auf ersten, zweiten und dritten Platz (3)

Gestartet wird das Programm mit XEQ "DERBY" , worauf Sie der Computer zur Eingabe einer Anfangszahl auffordert. Danach wird die Pferdenummer, der Pferdename sowie wie das jeweilige Pferd gesetzt ist, ausgedruckt. Danach können die Wetten plaziert werden. Zuerst wird die Pferdenummer (1...8), dann der Einsatz (...800 $) und schließlich die Position (1...3) eingegeben. Sind schließlich alle Wetten plaziert, wird bei der Aufforderung "PFERD NR. ?" einfach R/S gedrückt. Daraufhin erfolgt in 1 Minute der Start. Danach werden die Positionen der Pferde auf 5 Stationen ausgegeben. Außerdem wird ausgegeben wieviele Pferdelängen die jeweiligen Pferde hinter dem führenden liegen. Nach dem Zieleinlauf werden die Wetten ausbezahlt und der gesamte Gewinn bzw. Verlust berechnet.



L A B Y R I N T H

Der Spieler steht am Eingang eines Labyrinths. Dieses 25x10 m große Labyrinth soll er auf dem kürzesten Weg durchqueren. Um einen Vergleich zu erhalten, wie gut der Spieler war, hat der Computer das Lab. schon vorher einmal durchgangen, und der Spieler kann seine Wegstrecke mit der des Comp. vergleichen.

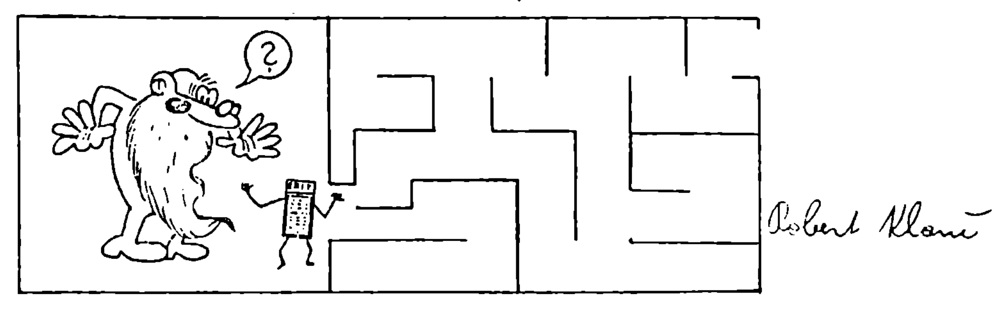
Gestartet wird das Programm mit XEQ"LAB" . Danach wird eine Anfangszahl und die Spielstufe eingegeben. Je größer die Spielstufe, desto weniger "Mauerstücke" blockieren den Weg. Danach sucht sich der Comp. ein Labyrinth Die lange Laufzeit des Aussuchens ist leider der Nachteil des Programmes. Dann geht der Comp. selbst durch das Labyrinth. Der wahre Grund warum er dies tut ist folgender: Bei dem Lab., welches der Comp. sich aussucht, kann es vorkommen, daß es keinen Weg hindurch gibt. Wenn jedoch der Comp. einmal durch das Lab. geht, und alle Mauersteine die ihm in den Weg geraten niederwalzt, gibt es sicher einen Weg durch das Lab.,nämlich den des Comp.. Deshalb ist es auch sehr schwierig die Wegstrecke des Comp. zu unterbieten.

Der Eingang des Labyrinths ist bei der Koordinate 1/5. Auf die Aufforderung "ZUG <X,Y> " (X,Y ist die momentane Positon) wird eine Richtungsanweisung eingegeben:

6..........nach rechts
4..........nach links
8..........nach oben
2..........nach unten

Ist der Weg in die eingegebene Richtung frei, so hat der Spieler 1m zurückgelegt. Ist der Weg blockiert, so erscheint "XX-STOP-XX", und der Spieler hat einen halben Meter zurückgelegt, ohne wirklich weitergekommen zu sein. Hat der Spieler den Ausgang (X-Koordinate 26) erreicht, kann er die Wegstrecken vergleichen. Will er jetzt noch das Lab. sehen, so kann er es mit "PRLAB" ausdrucken lassen.

Will der Spieler noch einmal von vorne anfangen (weil er keinen Ausgang fand)., so kann er dies mit XEQ A.



Barchewitz
Wasserwerkstr.33
68 Mannheim 31

10.5.81

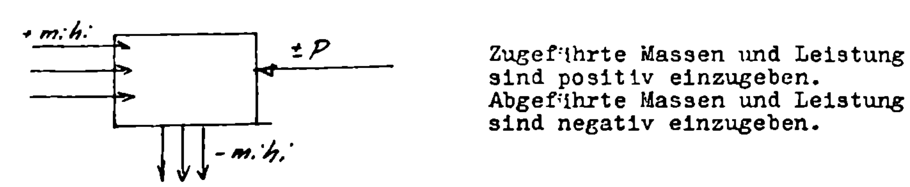
PROGRAMM "BIL"

Zweck: Ermittelt aus einem System, welches im thermodynamischen Gleichgewicht steht und welches aus max. 10 Massenströmen (m_i) mit den Wärmeinhalten (h_i) besteht, sowie einer zu- bzw. abgeführten Leistung (P), jeweils zwei unbekannte Größen.

D.M.: - m_1 und m_2 (no Flag)
- m_1 und P (SF01)
- m_1 und h_1 (SF02)

Anwendungsbeispiele: Thermodynamische Komponenten von Chemieanlagen, Lüftungsanlagen, Dampfkraftwerken etc.

Berechnungsansatz: Thermodynamisches System im Gleichgewicht.



Zugeführte Massen und Leistung sind positiv einzugeben.
Abgeführte Massen und Leistung sind negativ einzugeben.

Massenbilanz $\sum_{i=1}^{10} m_i = 0$

Energiebilanz $P + \sum_{i=1}^{10} m_i h_i = 0$

Auflösung nach m_1, m_2: (CF01, CF02)

$m_1 = -\sum_{i=3}^{10} m_i - m_2$ $m_2 = \left(\sum_{i=3}^{10} m_i h_1 - \sum_{i=3}^{10} m_i h_i - P\right) \cdot 1/(h_2 - h_1)$

Auflösung nach P, m_1: (SF01)

$m_1 = -\sum_{i=2}^{10} m_i$ $P = -\sum_{i=2}^{10} m_i h_i - m_1 h_1$

Auflösung nach m_1, h_1: (SF02)

$m_1 = -\sum_{i=2}^{10} m_i$ $h_1 = \left(-P - \sum_{i=2}^{10} m_i h_i\right) \cdot 1/m_1$

Dimensionen: Diese können frei gewählt werden. Sie müssen nur zueinander passen.

Z.B.: Masse in kg/s
Wärmeinhalt in kJ/kg
Leistung in KW

Speicherbelegung: Erforderlich ist SIZE 033!

```
STO oo   P
    01   m1
     :
    10   m10
    11   h1
     :
    20   h10
    21 ┐
    22 │ Hilfsspeicher
    23 ┘
    24   Anzahl der mi
    25 ┐  m1,m1²,h1,h1²,h1m1,n...in ΣREG
     : │
    31 ┘
    32   Hilfsspeicher
```

Programmanwendung:

0. Size 033
1. Magnetkarten laden. (5 Spuren)
2. XEQ BIL — Der Rechner zeigt die drei Lösungsansätze
   - -NO FLG M1,M2
   - R/S -SF 01 P,M1
   - R/S -SF 02 H1,M1
3. Gewünschtes Flag setzen bzw. R/S falls Rechner bereits im richtigen Status.
4. A (im USER-MODE): Eingabe neuer bzw. Überprüfung alter Werte
5. E " : Falls die Eingabe komplett bis zum letzten Element erfolgt, wird automatisch gerechnet
6. D " : Ausgabe der Ergebnisse erfolgt automatisc nach E.
7. a " : Falls Drucker angeschlossen, werden alle Werte incl. der Ergebnisse ausgedruckt.

Weiterentwicklung des Programms:

Mir schwebt vor Programm BIL zur Berechnung kompletter Kreisläufe zu erweitern. Konkret denke ich an Kraftwerkskreisläufe mit Kesseln, Turbinen, Speisewasservorwärmung etc

Hierzu wäre aus meiner Sicht erforderlich:
- Kreislaufbeschreibung mit Kennzahlen
- Vorgabe einer Rechenfolge für die einzelnen Komponenten.
- Vorgabe eines Iterationszieles
- Unterprogramm zum Umspeichern der Daten einer Komponente in den Arbeitsbereich von BIL

Wer hat Interesse?

```
 01+LBL "DERBY"
CLRG  FIX 0  CF 25
SF 12  "0<X<1"  PROMPT
STO 00  ADV  " DERBY"
PRA  .02

 13+LBL 10
CF IND X  ISG X  GTO 10
1.008  STO 09

 19+LBL 09
16  XEQ 50  INT  1  X<>Y
X<=Y?  2  FS? IND X
GTO 09  SF IND X
STO IND 09  ISG 09
GTO 09  2.008  STO 09

 35+LBL 12
RCL 09  STO 10  1  -
RCL IND X

 41+LBL 13
RCL IND 10  X<Y?  X<>Y
STO IND 10  RDN  ISG 10
GTO 13  STO IND Y
ISG 09  GTO 12  1.008
STO 09  CF 12  CF 13

 56+LBL 16
RCL 09  ACX  XEQ IND 09
"  "  ARCL IND 09  "├:1"
ACA  PRBUF  ISG 09
GTO 16  ADV
"POSITION: "
"├1,2 ODER 3"  PRA
"EINSATZ: "  "├BIS 800$"
PRA  20.029  STO 09

 76+LBL 18
0  "PFERD NR. ?"  PROMPT
X=0?  GTO 17  STO IND 09
"EINSATZ ?"  PROMPT
RCL 09  19  -  "WETTE "
ARCL X  "├:"  PRA  RDN
CLA  ARCL X  "├$ AUF"
ACA  RCL IND 09
XEQ IND X  X<>Y  1 E3  /
FRC  ST+ IND 09  40
ACCHR  "POSITION ?"
PROMPT  3  X<Y?  3  CLA
ARCL Y  ACA  X<>Y  1 E4
/  ST+ IND 09  41  ACCHR
PRBUF  ISG 09  GTO 18

123+LBL 17
ADV  SF 00  "START IN 1"
"├ MINUTE"  PRA  1.008
STO 09

131+LBL 00
10  RCL 09  +  LASTX
INT  1 E6  /  STO IND Y
ISG 09  GTO 00  21.025
STO 09

144+LBL 27
11.018  STO 10  FIX 1

148+LBL 19
4  XEQ 50  4  XEQ 50  +
2  /  RCL IND 10  10  *
FRC  1 E5  *  X<>Y
RCL IND Y  LOG  +  RND
ST+ IND 10  ISG 10
GTO 19  12.018  STO 19

172+LBL 26
RCL 19  STO 10  1  -
RCL IND X

176+LBL 14
RCL IND 10  X<Y?  X<>Y
STO IND 10  RDN  ISG 10
GTO 14  STO IND Y
ISG 19  GTO 26
"DAS RENNEN IST "
"├GESTARTET"  FS?C 00
PRA  11.018  STO 10
RCL 11  RND

197+LBL 11
ST- IND 10  ISG 10
GTO 11  ADV  XEQ IND 09
"POS. PFERD     "
"├ DAHINTER"  PRA  1.008
STO 10  FIX 0

209+LBL 15
RCL 10  ACX  10  +
RCL IND X  ENTER↑
ENTER↑  10  *  FRC  1 E5
*  XEQ IND X  FIX 1  RDN
RND  "  "  X≠0?  ARCL X
ACA  PRBUF  FIX 0
ISG 10  GTO 15  ISG 09
GTO 27  ADV  11.018
STO 09

239+LBL 29
RCL IND 09  10  *  FRC
1 E5  *  RCL 09  10  -
INT  10  /  ST+ IND Y
ISG 09  GTO 29  0
STO 12  20.029  STO 09

259+LBL 28
FIX 0  RCL 09  19  -
"WETTE "  ARCL X  "├:"
RCL IND 09  X=0?  GTO 32
PRA  INT  STO 10
RCL IND 09  FRC  1 E3  *
INT  STO 19  ST- 12
RCL IND 09  1 E3  *  FRC
10  *  INT  STO 11
RCL IND 10  FRC  10  *
X<=Y?  GTO 31
"ZERREISSE DEN "
"├WETTSCHEIN"  PRA

297+LBL 32
ISG 09  GTO 28  FIX 0
ADV  "DU "  RCL 12  X>0?
"├GEWANNST "  X<=0?
"├VERLORST "  "├INSG. "
ABS  ARCL X  "├$"  PRA
STOP

314+LBL 31
FIX 1  RCL IND 09
RCL 19  RCL IND Y  INT
*  RCL 11  /  ST+ 12
"DU KASSIERST "  ARCL X
"├$"  PRA  GTO 32

329+LBL 50
RCL 00  9821  *  .211327
+  FRC  STO 00  *  RTN

339+LBL 01
" MAN O WAR    "  ACA
RTN

343+LBL 02
" CITATION     "  ACA
RTN

347+LBL 03
" WHIRLAWAY    "  ACA
RTN

351+LBL 04
" DERBYSTAR    "  ACA
RTN

355+LBL 05
" SEABISCUIT   "  ACA
RTN

359+LBL 06
" GALLANT FOX  "  ACA
RTN

363+LBL 07
" ASSAULT      "  ACA
RTN
```

```
367◆LBL 08
" COALTOWN    " ACA
RTN

371◆LBL 21
" KURZ NACH "
"├DEN START" PRA RTN

376◆LBL 22
" NACH EINER "
"├1/4 MEILE" PRA RTN

381◆LBL 23
" NACH DER "
"├HALBEN DISTANZ" PRA
RTN

386◆LBL 24
" VOR DER "
"├ZIELGERADE" PRA RTN

391◆LBL 25
" ZIELEINLAUF" PRA
.END.
```

```
01◆LBL "LAB"
CLRG CF 21 "0<X<1"
PROMPT STO 00
"STUFE (1-6)?" PROMPT
6 X<=Y? 6 X<>Y 10 /
1.4 + STO 28 25
STO 26

20◆LBL 00
9 E-3 STO 27

23◆LBL 12
RCL 28 XEQ 10 RCL 27
INT 10↑X * ST+ IND 26
ISG 27 GTO 12 DSE 26
GTO 00 GTO 15

36◆LBL 10
RCL 00 9821 * .211327
+ FRC STO 00 * INT
RTN

47◆LBL 15
CF 00 CF 01 CF 02 1
STO 30 5 STO 31 ADV
2 STO 32 XEQ 98

59◆LBL 16
3 XEQ 10 20 +
XEQ IND X XEQ 98 10
RCL 31 X=Y? SF 01 1
ST+ 34 X=Y? SF 02
GTO 16

75◆LBL A
CF 21 1 STO 30 5
STO 31 3 STO 32
XEQ 98 BEEP 0 STO 26
STO 33 FIX 0 CF 29

90◆LBL B
"ZUG ? (" ARCL 30 "├,"
ARCL 31 "├)" CF 22
AVIEW

98◆LBL 05
PSE FC?C 22 GTO 05 2
/ FRC X≠0? GTO B
LASTX XEQ IND X XEQ 98
1 ST+ 33 TONE 9 GTO B

114◆LBL 01
1 ST- 31 XEQ 09 X≠0?
RTN 1 ST+ 31 GTO 11

123◆LBL 02
1 ST- 30 XEQ 09 X≠0?
RTN 1 ST+ 30 GTO 11

132◆LBL 03
1 ST+ 30 XEQ 09 X≠0?
RTN 1 ST- 30 GTO 13

141◆LBL 04
1 ST+ 31 XEQ 09 X≠0?
RTN 1 ST- 31 GTO 11

150◆LBL 09
RCL IND 30 RCL 31 10↑X
/ FRC 10 * INT RTN

160◆LBL 20
1 FC?C 00 SF 00
ST+ 30 RCL 30 26 X=Y?
GTO A FS? 00 RTN
CF 01 CF 02 RTN

174◆LBL 21
FS? 02 GTO 16 1
ST- 31 SF 01 CF 00
RTN

182◆LBL 22
FS? 01 GTO 16 1
ST+ 31 SF 02 CF 00
RTN

190◆LBL 13
25 RCL 30 X=Y? GTO 26

195◆LBL 11
"***-STOP-***" AVIEW
TONE 0 .5 ST+ 33
GTO B

202◆LBL 98
RCL IND 30 RCL 31 10↑X
/ FRC 10 * INT
RCL 31 1 - 10↑X *
ST- IND 30 RCL 32
LASTX * ST+ IND 30
RTN

222◆LBL 08
223◆LBL "PRLAB"
SF 12 SF 21 ADV
"*...SPIELER" PRA 10
ACCHR "...COMPUTER"
ACA PRBUF ADV
"EINGANG" PRA 5
SKPCHR 7 ACCHR PRBUF
" ----------" PRA
1.025 STO 27

246◆LBL 06
1.01 STO 28 124 ACCHR

251◆LBL 07
RCL 27 INT STO 30
RCL 28 INT STO 31
XEQ 09 60 + XEQ IND X
ACCHR ISG 28 GTO 07
124 ACCHR PRBUF
ISG 27 GTO 06 PRA
STOP

272◆LBL 60
31 RTN

275◆LBL 61
32 RTN

278◆LBL 62
10 RTN

281◆LBL 63
42 RTN

284◆LBL 26
TONE 4 TONE 7 TONE 5
TONE 4 TONE 6 TONE 5
TONE 4 TONE 7 TONE 6
TONE 4 TONE 7 TONE 4
ADV SF 21 SF 12
"WEGSTRECKE:" AVIEW
"COMPUTER " ARCL 34
"├M" AVIEW "SPIELER "
ARCL 33 "├M" AVIEW
AON "PLOT J-N?" PROMPT
AOFF ASTO X "J"
ASTO Y X=Y? GTO 08
.END.
```

Programm 3R

```
01♦LBL "BIL
       "
02 SF 27
03 CLA
04 "NO FLG*
   M1,M2"
05 PROMPT
06 CLA
07 "SF01*P,
       M1"
08 PROMPT
09 CLA
10 "SF02*H1
      ,M1"
11 PROMPT
12 GTO A

13♦LBL 20
14 ΣREG 25
15 CLΣ
16 XEQ 04
17 RCL 21
18 10,013
19 +
20 STO 22

21♦LBL 01
22 RCL IND
        22
23 RCL IND
        21
24 Σ+
25 1
26 ST+ 22
27 ISG 21
28 GTO 01
29 RTN

30♦LBL 06
31 RCL 25
32 RCL 11
33 *
34 RCL 00
35 -
36 RCL 29
37 -
38 RCL 12
39 RCL 11
40 -
41 /
42 STO 02
43 RCL 25
44 CHS
45 RCL 02
46 -
47 STO 01
48 RTN

49♦LBL 04
50 2
51 FS? 01
52 GTO 24
53 FS? 02
54 GTO 24
55 3

56♦LBL 24
57 STO 23
58 RCL 24
59 1 E3
60 /
61 RCL 23
62 +
63 STO 21
64 RTN

65♦LBL 07
66 RCL 25
67 CHS
68 STO 01
69 RTN

70♦LBL 08
71 RCL 29
72 CHS
73 RCL 01
74 RCL 11
75 *
76 -
77 STO 00
78 RTN

79♦LBL 09
80 RCL 00
81 CHS
82 RCL 29
83 -
84 RCL 01
85 /
86 STO 11
87 RTN

88♦LBL E
89 FS? 01
90 GTO 10
91 FS? 02
92 GTO 11
93 XEQ 20
94 XEQ 06
95 GTO D

96♦LBL 10
97 XEQ 20
98 XEQ 07
99 XEQ 08
100 GTO D

101♦LBL 11
102 XEQ 20
103 XEQ 07
104 XEQ 09

105♦LBL D
106 TONE 1
107 SF 21
108 CLA
109 "M1="
110 ARCL 01
111 RCL 01
112 XEQ IND
        32
113 FS? 02
114 GTO 13
115 FS? 01
116 GTO 14
117 CLA
118 "M2="
119 ARCL 02
120 RCL 02
121 XEQ IND
        32
122 GTO 19

123♦LBL 13
124 CLA
125 "H1="
126 ARCL 11
127 RCL 11
128 XEQ IND
        32
129 GTO 19

130♦LBL 14
131 CLA
132 "P="
133 ARCL 00
134 RCL 00
135 XEQ IND
        32
136 RTN

137♦LBL 19
138 CLA
139 "  ENDE"
140 XEQ IND
        32
141 STOP

142♦LBL A
143 FS? 00
144 GTO "AA"
145 25
146 STO 32
```

```
147◆LBL "AA"
148 "ANZ="
149 RCL 24
150 ARCL 24
151 XEQ IND
          32
152 STO 24
153 FS? 00
154 GTO B
155 CLA
156 "....XX"
157 ASTO 01
158 FS? 01
159 GTO 22
160 FS? 02
161 GTO 23
162 ASTO 02
163 GTO B

164◆LBL 22
165 ASTO 00
166 GTO B

167◆LBL 23
168 ASTO 11

169◆LBL B
170 XEQ 14
171 STO 00
172 RCL 24
173 1 E3
174 /
175 1
176 +
177 STO 21
178 10,01
179 +
180 STO 22

181◆LBL 15
182 XEQ 18
183 1
184 ST+ 22
185 ISG 21
186 GTO 15
187 FS?C 00
188 GTO 19
189 GTO E
190 RTN

191◆LBL 18
192 "M"
193 FIX 0
194 ARCL 21
195 "⊢="
196 FIX 2
197 ARCL IND
          21
198 RCL IND
          21
199 XEQ IND
          32
200 STO IND
          21
201 CLA
202 "H"
203 FIX 0
204 ARCL 21
205 "⊢="
206 FIX 2
207 ARCL IND
          22
208 RCL IND
          22
209 XEQ IND
          32
210 STO IND
          22
211 RTN

212◆LBL 25
213 PROMPT
214 RTN

215◆LBL 26
216 XEQ "PRA"
217 RTN

218◆LBL a
219 26
220 STO 32
221 SF 00
222 CLA
223 SF 12
224 "PROTOKOLL"
225 XEQ "ACA"
226 ADV
227 ADV
228 CF 12
229 "ADV"
230 XEQ A
231 RTN
232 END
```

Beispiel Eingabe:

```
           XEQ A
ANZ=3,00
             RUN
P=0,00
             RUN
M1,=....XX
             RUN
H1,=270,00
 280,00        R
              UN
M2,=....XX
             RUN
H2,=2.750,00
 2.740,00
             RUN
M3,=25,00
-25,00        RU
               N
H3,=2.400,00
             RUN
```

Eingabe

```
M1=3,46
             RUN
M2=21,54
             RUN
  ENDE
```

Ausgabe

```
           XEQ a
       PROTOKOLL

ANZ=3,00
P=0,00
M1,=3,46
H1,=260,00
M2,=21,54
H2,=2.740,00
M3,=-25,00
H3,=2.400,00
  ENDE
```

P R O G R A M M I E R H I L F E N   F Ü R   A N F Ä N G E R

Der HP-41 C(V) enthält einige Funktionen, deren praktischer Zweck nicht jedem Benutzer sofort klar ist; hierzu zählen u.a. RND, MOD und SIGN. Die Anwendung dieser Funktionen soll im folgenden an Beispielen gezeigt werden.

RND (Seite 78 des deutschen Hanbuchs)

RND rundet eine im X-Register stehende Zahl entsprechend dem eingestellten Anzeigeformat, und zwar nicht nur in der Anzeige sondern auch für alle folgenden Rechnungen mit dieser Zahl. Diese Eigenschaft läßt sich zunächst einmal bei der Rechnung mit Mark und Pfennig anwenden, wenn Einzelpreise auch noch Zehntelpfennige enthalten (z.B. die auf Kundenfang ausgelegten 9/10 Pfennige der Tankstellen), man aber von der Endsumme ab nur mit ganzen Pfennigen weiterarbeiten will.

Ein zweiter wichtiger Zweck der RND-Funktion liegt in der Verarbeitung kleiner Zahlen. Da ist zunächst die Anzeige von Zahlen, die kleiner als das eingestellte Anzeigeformat sind. Bekanntlich schaltet der HP-41 C(V) beispielsweise bei Zahlen, die betragsmäßig kleiner als 0.005 sind, automatisch von FIX 2 auf SCI 2 um, damit auch diese Zahlen noch angezeigt werden. Das ist aber häufig vom Benutzer unerwünscht, da dann evtl. die 12 Stellen des Displays überschritten oder die Lesbarkeit einer Tabelle gestört werden. Führt man jedoch vor der entsprechenden Anzeigefunktion (PROMPT, R/S, PRX oder VIEW__) oder vor der Übernahme einer Zahl ins ALPHA-Register (ARCL__) RND aus, so erscheint die betreffende Zahl als 0.00.

Die Eigenschaft, daß die Rundung nicht nur in der Anzeige erfolgt, sondern daß die gerundete Zahl auch für weitere Rechnungen zur Verfügung steht, kann man sich vorteilhafterweise bei Iterationsrechnungen zunutze machen, wenn man die Iterationsgenauigkeit durch das Anzeigeformat festlegen möchte. Bei der Iterationsrechnung zur Bestimmung einer Zahl x wird häufig von einem irgendwie bestimmten Näherungswert x_n ausgegangen, der durch Anbringen einer kleinen Korrektur Δx_n zu einem neuen Näherungswert $x_{n+1} = x_n + \Delta x_n$ verbessert wird. So ist beim

Programmierhilfen für Anfänger (Fortsetzung)

Newton-Verfahren zur Bestimmung einer Nullstelle x der Funktion f(x) diese Korrektur durch $\Delta x_n = -f(x_n)/f'(x_n)$ gegeben. Dieses Verfahren wird fortgesetzt, bis die Korrektur Δx_n betragsmäßig unter einen kleinen vorgegebenen Wert ε gesunken ist. Am besten wäre es natürlich, die Iteration dann abzubrechen, wenn $\Delta x_n = 0$ ist. Das ist aber -nicht zuletzt wegen der Rundungsfehler- so gut wie nie zu erreichen. Man kann jedoch Δx_n im Rahmen der gewünschten Anzeigegenauigkeit gewaltsam zu Null machen, indem man vor die Abfrage X=0? ein RND setzt; in diesem Falle ist sogar die vorherige Betragsbildung von Δx_n durch ABS überflüssig.

MOD (Seite 79 des deutschen Handbuchs)

MOD ermittelt den Rest, der bei der Division Y/X bleibt. Abgesehen von dieser einfachen Aufgabe läßt sich die MOD-Funktion aber auch in Programmen für andere Zwecke verwenden, was hier an zwei Beispielen demonstriert werden soll:

Im ersten Beispiel soll eine Tabelle ausgedruckt werden, bei der die Lesbarkeit durch Einfügen einer Leerzeile nach jeder fünften Druckzeile erhöht werden soll. Nimmt man an, daß sich in R00 ein Zeilenzähler befindet, der vor Druckbeginn auf Null gesetzt ist und nach dem Ausdruck einer Zeile jeweils um 1 erhöht wird (1 ST+00 oder mit ISG 00, dann muß vor Anwendung von MOD ein INT gesetzt werden), so läßt sich dieses Problem mit RCL 00 (INT) 5 MOD X=0? ADV lösen: nach jeweils 5 ausgedruckten Zeilen ist der Zeilenzähler ohne Rest durch 5 teilbar, d.h. in diesem Falle steht nach der Ausführung von MOD im X-Register eine Null.

Das zweite Beispiel führt in die Navigation: alle irgendwie errechneten Azimutwinkel (Kurs, rechtweisende Peilung etc.) sollen zwischen 0^0 und 360^0 liegen; sie liegen jedoch nach der Rechnung, da sie meist durch Anwendung von $\sin^{-1}$, $\cos^{-1}$, $\tan^{-1}$ oder R-P entstehen, zwischen -180^0 und $+180^0$. Nimmt man an, daß der betreffende Winkel im X-Register steht, so wird er durch 360 MOD ohne weitere Entscheidungen und Sprungbefehle in die "vollkreisige" Anzeige umgewandelt.

Programmierhilfen für Anfänger (Fortsetzung)

SIGN (Seite 85 des deutschen Handbuchs)

In der Mathematik wird die Signum-Funktion folgendermaßen definiert:

$$\mathrm{sign}(x) = \begin{cases} -1 & \text{für } x < 0 \\ 0 & \text{für } x = 0 \\ +1 & \text{für } x > 0 \end{cases}$$

Die Definition im HP-41 C(V) weicht hiervon etwas ab, indem +1 für $x \geq 0$ ins X-Register geschrieben wird, 0 jedoch dann, wenn der Inhalt des X-Registers aus ALPHA-Zeichenbesteht. Hieraus ergibt sich bereits die erste Anwendungsmöglichkeit: man kann nämlich ohne ALPHA ERROR prüfen, ob ein Register Zahlen oder ALPHA-Zeichen (oder als solche "getarnte" Zahlen) enthält: RCL__ SIGN X=0?

Häufig möchte man den Wert, den eine Variable annehmen kann, symmetrisch zur Null-Linie auf $\pm L$ begrenzen. Nimmt man an, daß die betreffende Variable im X-Register steht, so bringt SIGN LAST X ABS L X<Y? X<>Y RDN * den gewünschten Begrenzungseffekt: SIGN rettet das Vorzeichen der Variablen über die nachfolgende Betragsbildung hinweg, ohne daß zeitraubende Sprünge zu Labels gemacht werden müssen. In trigonometrischen Rechnungen muß gelegentlich $\sin^{-1}x$ oder $\cos^{-1}x$ gebildet werden; hierfür ist $-1 \leq x \leq +1$ Voraussetzung. Durch Rundungseffekte kann x aber bei längeren Rechnungen diese Grenzen geringfügig überschreiten. In diesem Falle hilft vor Anwendung von SIN^{-1} oder COS^{-1} die obige Befehlsfolge mit L = 1.

Schließlich eignet sich SIGN auch gut zur Erzeugung einer periodischen Rechteckfunktion: $f(x) = \mathrm{sign}(\sin x)$.

In den bisher gebrachten Beispielen störte die im HP-41 C(V) verwendete, unübliche Definition von sign(x) nicht. Sollte man jedoch einmal die in der Mathematik übliche Definition explizit benötigen, so ist das mit X≠0? SIGN leicht zu bewerkstelligen.

Klaus Werner Hoenow (11)

Kampweg 7a
1000 Berlin 27

Automatische Anzeige des Zustandes sämtlicher Flags.

Für alle Freunde des HP - 41C(V), die noch ohne Drucker auskommen müssen, ist es etwas mühsam, wenn viele oder alle Flags abgefragt werden sollen.
Das Programm von Rüdiger Schultz (95-81) ist da schon eine gute Sache, aber es war mir zu lang!

Nachfolgend meine bisher kürzeste Version:
Das Programm zeigt nur die gesetzten Flags an und ist deshalb sehr schnell.
Wer so schnell nicht gucken kann, sollte hinter Zeile 10 noch ein PSE einbauen, ich persönlich halte es für unnötig.

Dieses Programm, natürlich ohne PSE, ist auch für die Benutzung mit Drucker sehr interessant, weil es mit Papier und Zeit viel sparsamer umgeht als 'PRFLAG

Falls das Programm öfter gebraucht wird, sollte man es der Taste -54 (FS?) zuordnen!

```
W. STROINSKI 12. 8. 81

ANZEIGE ODER DRUCK DER
GESETZTEN FLAGS.

MEIST SCHNELLER ALS
"PRFLAGS".

01♦LBL "F?"
02 ,055
03♦LBL 00
04 INT
05 FC? IND X
06 GTO 00
07 CLA
08 ARCL X
09 "⊢SET"
10 AVIEW
11♦LBL 00
12 LASTX
13 ISG X
14 GTO 00
15 END

35 BYTES
INHALTE VON X, Y, Z, T
UND L GEHEN VERLOREN.
```

Werner Stroinski
Kampweg 7a
1000 Berlin 27

Berlin-Tegel, 13. 8. 81

Schutz der Magnetkarten-Beschriftung.

Beschriftung der Magnetkarten mit Bleistift oder Tusche ist häufig den Beanspruchungen nicht gewachsen und wird im Laufe der Zeit verwischt und unansehnlich.
Der Schutz derartiger Beschriftungen läßt sich mit den üblichen Sprühlacken durchführen (Marabu, Pelikan u. ä.), aber... was passiert, wenn auch nur ein Tröpfchen auf die Magnetseite gerät? MALFUNCTION... CHECKSUM ERROR,, usw...

Einen einfachen Schutz habe ich ursprünglich mit einfachem TESA-Film erzielt, den ich auf die Magnetseite klebte, um ihn nach dem Lackieren wieder zu entfernen. Das geht aber nur bei den inzwischen selten gewordenen alten Magnetkarten gut, die man leicht an der 45°Kerbe erkennen kann. Bei den neueren Karte verbindet sich der Kleber des TESA-Films mit der Magnetschicht so "innig", daß man große Mühe hat, wieder lesbaren Zustand herzustellen.

Eine einwandfreie Schutzmöglichkeit der Magnetschicht aller Karten besteht dari die Magnetschicht mit einem Abdecklack (Humbrol-Maskol) zu schützen. Dieser Lack (Modellbau-Artikel!) stinkt zwar und sieht auch sonst wenig vertrauenserweckend aus, er schützt aber einwandfrei und läßt sich am Schluß der Behandlung mühelos von der Magnetschicht abziehen oder abrollen, da er dann gummiähnliche Eigenschaften hat.

Umwandlun‾von maximal 8-s
(Bytes) in 2-stellige Hexade

Programm	Kommentar
01◆LBL "HX" 02 CF 12 03 SF 27 04 CF 28 05 CF 29 06 FIX 0	Haupt-Label Status- Einstellung
07 ,009 08◆LBL 00 09 INT 10 STO IND X 11 RDN 12 LASTX 13 ISG X 14 GTO 00	Speichern der Hex-Ziffern 0 bis 9 in den Registern 00 bis 09
15 RDN 16 "A" 17 ASTO 10 18 "B" 19 ASTO 11 20 "C" 21 ASTO 12 22 "D" 23 ASTO 13 24 "E" 25 ASTO 14 26 "F" 27 ASTO 15	Stack ordnen, (evtl. erfolg-te Eingabe retten). Speichern der Hex-Ziffern A bis F in den Registern 10 bis 15
28 FC?C 22 29 STOP	Stop nur, wenn keine Eingabe erfolgte!
30◆LBL A 31 SF 21 32 ADV	Lokales Label Anzeige/Druck-Modalitäten
33 CLA 34 ARCL X 35 "⊢ " 36 FS? 55 37 "⊢BIN=HEX "	Anzeige/Druck Vorbereitungen
38 CF 00 39 1 E4 40 / 41 FRC 42 STO 17	1. Durchlauf kennzeichnen, Abtrennen, speichern der 4 niedrigsten Bit für später
43 LASTX 44 INT 45 X≠0? 46 GTO 05 47 ARCL 00 48 GTO 02	Abtrennen der 4 höchsten Bits Prüfung, ob von 0 verschieden, sonst als Hex-Ziffer 0 aufrufen!
49◆LBL 05 50 0 51 STO 16 52 RDN	Sammel-Reg. löschen.
53 1 E3 54 X>Y? 55 GTO 01 56 - 57 8 58 ST+ 16	Untersuchung, ob 1. bzw. 5. Bit den Wert 0 oder 1 hat,im letzten Fall 8 speichern!
59◆LBL 01 60 RDN 61 100 62 X>Y? 63 GTO 01 64 - 65 4 66 ST+ 16	Untersuchung des 2. bzw. 6. Bits, 4 addieren, falls dieses Bit≠ 0 war.
67◆LBL 01 68 RDN 69 10 70 X>Y? 71 GTO 01 72 - 73 2 74 ST+ 16	Untersuchung des 3. bzw. 7. Bits, 2 addieren, falls dieses Bit≠ 0 war.
75◆LBL 01 76 RDN 77 ST+ 16	Wert des 4. bzw. 8. Bits addieren.
78 X=0? 79 GTO 04 80 1 81 X=Y? 82 GTO 04	Prüfung des 4. bzw. 8. Bits auf 0 oder 1, falls anderer Rest bleibt, Eingabe falsch
83 "⊢?" 84 GTO 03	Fehler-Symbol vorbereiten!
85◆LBL 04 86 ARCL IND 16	Hex-Ziffer aufrufen,anfügen.
87◆LBL 03 88 FC? 00 89 GTO 02	Wurde 1.bis4. Bit bearbeitet?
90 FC? 55 91 AVIEW 92 ACA 93 ADV 94 RTN	Fehlt Drucker? Anzeigen! Buffer laden, drucken! Programm-Stop!

stelligen Binärzahlen
zimalzahlen.

```
 95+LBL 02
 96 RCL 17
 97 1 E4
 98 *
 99 SF 00
100 GTO 05
101 END

182 BYTES
TOT. REG. 44
SIZE 018

USER MODUS:
A

BEISPIELE:

        0 BIN=HEX 00

        1 BIN=HEX 01

      111 BIN=HEX 07

      121 BIN=HEX 0?
FALSCHES ZEICHEN IM
2. HALBBYTE.

 10101010 BIN=HEX AA

 11111111 BIN=HEX FF

 12111010 BIN=HEX ?A
FALSCHES ZEICHEN IM
1. HALBBYTE.

 51101411 BIN=HEX ??
FALSCHE ZEICHEN IN
BEIDEN HALBBYTES.

100110111 BIN=HEX ?7
MEHR ALS 8 STELLEN, NUR
DIE 4 NIEDRIGSTEN WERDEN
BEARBEITET.

========================
```

Rückruf und Regeneration des 5. bis 8. Bits.

2. Durchlauf kennzeichnen.

Register-Inhalte:

00 :	0
01 :	1
02 :	2
03 :	3
04 :	4
05 :	5
06 :	6
07 :	7
08 :	8
09 :	9
10 :	"A"
11 :	"B"
12 :	"C"
13 :	"D"
14 :	"E"
15 :	"F"
16 :	Adresse für Hex-Ziffer
17 :	5. bis 8 Bit

Stroinski, Aug. 81 (82)

HX/A USER SIZE 018

Binär → Hexadezimal

Programm-Beschreibung:

Die eingegebene, aus den Ziffern 0 und 1 aufgebaute, maximal 8-stellige Binärzahl wird in 2 je 4-stellige Teile aufgetrennt, die jeweils eine Stelle der Hexadezimalzahl ergeben.
Dazu wird die 4-stellige Binärzahl als Dezimalzahl angesehen und nacheinander versucht, 1000 bzw. 100 bzw 10 abzuziehen, ohne ein negatives Ergebnis zu erhalten. Falls möglich, werden die Subtraktionen ausgeführt und der Stellenwert des bearbeiteten Bits im Register 16 aufsummiert. Der Wert des letzten Bits wird dieser Summe zugeschlagen und ergibt die Adresse der aufzurufenden Hex-Ziffer

Damit auch die korrekte Eingabe kontrolliert wird, ist beim niedrigsten Bit jeder Gruppe eine Prüfung auf 0 oder 1 vorgesehen, da nur diese Werte auftreten dürfen. Bei abweichenden Werten wird an die Stelle der Hex-Ziffer ein "?" eingefügt.

Programm-Benutzung:

1. Programm eingeben.
2. XEQ 'HX'
3. Binärzahl eingeben, dabei nur die Ziffern 0 und 1 verwenden!
4. XEQ 'A'
   (Bei erstmaliger Benutzung des Programms darf Schritt 3 auch vor Schritt 2 ausgeführt werden, Schritt 4 wird dann überflüssig)
5. Nächste Binärzahl eingeben, weiter bei 4.

Bei Benutzung ohne Drucker werden Binär- und Hexadezimal-Zahl nebeneinander, getrennt durch eine Lücke, angezeigt:

10101010 AA

Es können dann die Zeilen 93/92/90 37/36/32 ersatzlos gelöscht werden. Das Programm ist dann nur noch 164 Bytes lang und benötigt insgesamt nur noch 42 Register!

Andreas Stiens
Dammstraße 1
44oo Münster
43o

Münster,2o.o7.1981

Liebe Clubfreunde!

Die Behauptung Olivers aus Prisma 196,197-81,daß die Clubmitglieder mit höherer Nummer ziemlich passiv seien konnte ich nicht auf mir sitzen lassen. Fürs Erste schicke ich mal zwei Spielprogramme. Da wir uns ja im Zeitalter des Energiesparens befinden und das Autofahren immer teuerer wird, schicke ich zwei Autorennspiele.
Leider sind mir die Magnetkarten ausgegangen,ich habe gerade erst neue bei Ulli bestellt,so daß ich die beiden Spiele noch nicht auf Magnetkarten mitsenden kann.
In irgend einem Prisma (ich kann es nicht wiederfinden) wurde zum aufstellen von Vermessungsprogr. aufgerufen,ich habe schon eine ganze Anzahl zusammen,zu denen ich allerdings noch eine Erläuterung schreiben muß.Ich werde sie Oliver nach und nach zuschicken.

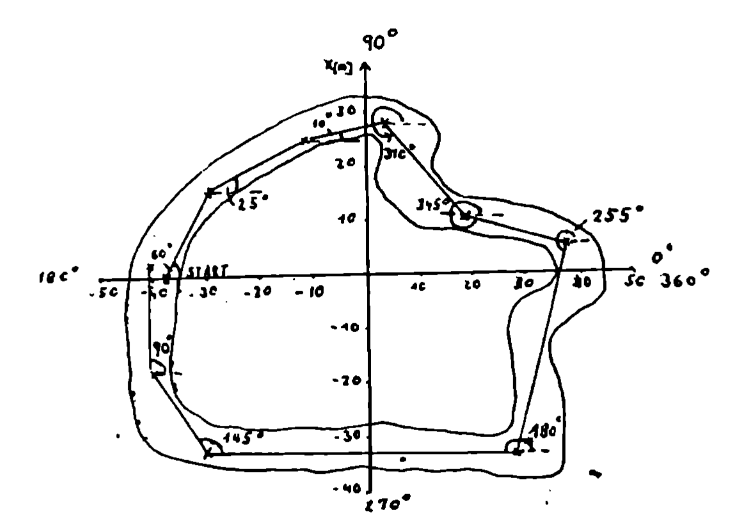
Autorennen nach Koordinaten
8o Prgm Reg.
SIZE o4o

```
oo1 LBL'R
oo2 CLRG
oo3 FIX 0
oo4'V/M.:7,14,21,28
oo5 PROMPT
oo6'X:_2,9,16,23
oo7 PROMPT
oo8'Y:_3,1o,17,24
oo9 AVIEW
o1o RTN
o11 LBL'AA
o12 1
o13 STO 36
o14 1,1
o15 GTO o2
o16 LBL'AB
o17 2
o18 STO 36
o19 8,1
o2o GTO o2
o21 LBL'AC
o22 3
o23 STO 36
o24 15,1
o25 GTO o2
o26 LBL'AD
o27 4
o28 STO 36
o29 22,1
o3o LBL o2
o31 STO 38
o32 29,o35
o33 STO 39
o34 LBL o3
o35 RCL 38
o36 INT
o37 STO IND 39
o38 ISG 38
o39 ISG 39
o4o GTO o3
o41 LBL'EF
o42 RCL IND 32
o43 STO IND 33
o44 'a,∡ ?
o45 PROMPT
o46 ISG IND 34
o47 1
o48 STO IND 32
o49 X<>Y
o5o 2
o51 /
o52 ST+ IND 29
o53 RCL IND 29
o54 RCL IND 35
o55 3,6
o56 /
o57 X∠=Y?
o58 STO IND 29
```

```
o59 RCL IND 32
o6o RCL IND 29
o61 P-R
o62 ST+ IND 31
o63 X <> Y
o64 ST+ IND 3o
o65 RCL IND 33
o66 18o
o67 +
o68 RCL IND 32
o69 -
o7o 3
o71 1/X
o72 *
o73 ABS
o74 RCL IND 29
o75 X > Y?
o76 GTO'UE
o77 XEQ'UN
o78 RCL IND 29
o79 3,6
o8o *
o81 'V=
o82 ARCL X
o83 '├_KM/H
o84 AVIEW
o85 PSE
o86 'Y=
o87 ARCL IND 31
o88 '├_X=
o89 ARCL IND 3o
o9o AVIEW
o91 RTN
o92 LBL'UE
o93 'WAGEN_
o94 ARCL 36
o95 '├_HAT_SICH
o96 AVIEW
o97 'UEBERSCHLAGEN
o98 AVIEW
o99 RTN
1oo LBL'UN
1o1 RCL o2
1o2 RCL o9
1o3 -
1o4 RCL o3
1o5 RCL 1o
1o6 XEQ o1
1o7 X > Y?
1o8 XEQ '1
1o9 RCL o2
11o RCL 16
111 -
112 RCL o3
113 RCL 17
114 XEQ o1
115 X > Y?
116 XEQ '2
117 RCL o2
118 RCL 23
119 -
12o RCL o3
121 RCL 24
122 XEQ o1
123 X > Y?
124 XEQ '3
125 RCL o9
126 RCL 16
127 -
128 RCL 1o
129 RCL 17
13o XEQ o1
131 X > Y?
132 XEQ '4
133 RCL o9
134 RCL 23
135 -
136 RCL 1o
137 RCL 24
138 XEQ o1
139 X > Y?
14o XEQ '5
141 RCL 17
142 RCL 24
143 -
144 RCL 18
145 RCL 25
146 XEQ o1
147 X > Y?
148 XEQ '6
149 RTN
15o LBL o1
151 -
152 R-P
153 1
154 RTN
155 LBL '1
156 '1_U._2
157 ASTO 37
158 GTO 'U
159 LBL '2
16o '1_U._3
161 ASTO 37
162 GTO 'U
163 LBL '3
164 '1_U._4
165 ASTO 37
166 GTO 'U
167 LBL '4
168 '2_U._3
169 ASTO 37
17o GTO 'U
171 LBL '5
172 '2_U._4
173 ASTO37
174 GTO 'U
175 LBL '6
176 '3_U._4
177 ASTO37
178 LBL'U
179 'WAGEN_
18o ARCL 37
181'├_SIND
182 AVIEW
183 'ZUSAMMEN
184 '├GESTOSSEN
185 AVIEW
186 RTN
187 LBL 'Z
188 CLA
189 ARCL o6
19o '├S_
191 ARCL 13
192 '├S_
193 ARCL 2o
194 '├S_
195 ARCL 27
196 '├S
197 AVIEW
198 END
```

Das Programm kann von 1-4 Personen gespielt werden.
Starten mit XEQ 'R
Vmax in km/h in R 7,14,21, 28 eingeben.
R/S
X-Koordinaten in R 2,9,16,23 eingeben
R/S
Y-Koordinaten in R 3,1o,17, 24 eingeben.

Wird mit weniger als vier Autos gespielt,so brauchen auch nur entsprechend weniger Daten eingegeben werden.Allerdings sollten für nicht benutzte Wagen die X-oder Y-Koordinate anders seien als bei jedem anderen Wagen,da der Rechner sonst laufend Zusammenstöße meldet.Das Gleiche gilt nach Unfällen,hier sollte auch eine Koordinate jedes am Unfall beteiligten Wagens geändert werden, da der Rechner den Unfall sonst immer wieder meldet.
Man zeichnet sich auf ein Blatt kariertes Papier ein Koordinatensystem,Maßstab des Koordinatensystems bestimmt die Länge des Rennkurses,und einen beliebigen Rennkurs.
Nun trägt man die Startkoordinaten in den Rennkurs ein und startet mit XEQ 'AA,AB,AC,AD je nachdem welcher Wagen benutzt werden soll.Der Rechner zeigt dann 'a,∠. a ist die Beschleunigung (negativ eingegeben Bremsung) und ∠ ist der Winkel bezogen auf den Einheitskreis.
Wird eine Kurve zu schnell durchfahren erscheint: 'WAGEN HAT SICH UEBERSCHLAGEN.
Kommen sich zwei oder mehrere Wagen zu nahe (∠ 1m) erscheint:WAGEN N U. M SIND ZUSAMMENGESTOSSEN
Hat ein Fahrzeug die am Anfang eingegebene Geschw. Vmax. erreicht;so kann zwar weiter eine positive Beschleunigung eingegeben werden,der Rechner hält den Wagen aber auf Vmax.
Die gefahrenen Zeiten können mit XEQ 'Z abgerufen werden,sie erscheinen inder Reihenfolge AA,AB,AC,AD!
Beispiel für einen Rennkurs,den ein Wagen durchfuhr, die eingegebenen Winkel sind richtig dargestellt.



Autorennen nach Anzeige

64 PRGM Reg.
SIZE 009

```
001 LBL 'AB
002 'V?
003 PROMPT
004 STO 01
005 FIX 0
006 CF 29
007 'AUGENBLICK
008 AVIEW
009 4
010 STO 03
011 CLX
012 STO 07
013 STO 06
014 LBL 11
015 CLX
016 STO 06
017 CF 01
018 CF 02
019 CLA
020 ASTO 05
021 2
022 XEQ 'RA
023 X=0?
024 SF 01
025 X≠0?
026 GTO 12
027 2
028 XEQ 'RA
029 X=0?
030 XEQ 20
031 X≠0?
032 XEQ 21
033 TONE 0
034 'KURVE
035 ARCL 08
036 AVIEW
037 450
038 XEQ 'RA
039 300
040 +
041 STO 02
042 LBL 12
043 4
044 XEQ 'RA
045 3
046 +
047 STO 00
048 LBL 00
049 RCL 01
050 ST+ 07
051 XEQ 'GA
052 FS? 01
053 XEQ 21
054 FS? 02
055 XEQ 20
056 5
057 XEQ IND 03
058 SF 00
059 AVIEW
060 PSE
061 AVIEW
062 CF 00
063 5
064 -
065 1,7
066 /
067 RND
068 ST+ 03
069 ST+ 06
070 FC? 01
071 GTO 08
072 RCL 02
073 RCL 01
074 R-P
075 RCL 02
076 -
077 FS? 02
078 CHS
079 ST+ 06
080 RCL 06
081 ST+ 03
082 LBL 08
083 1
084 RCL 03
085 X<=Y?
086 GTO 09
087 7
088 X<=Y?
089 GTO 10
090 DSE 00
091 GTO 00
092 GTO 11
093 LBL 09
094 X<>Y
095 LBL 10
096 STO 03
097 XEQ 'GA
098 FS? 01
099 XEQ 21
100 FS? 02
101 XEQ 20
102 XEQ IND 03
103 AVIEW
104 PSE
105 'UNFALL
106 AVIEW
107 BEEP
108 BEEP
109 'GEFAHREN:_
110 ARCL 07
111 '⊢_METER
112 AVIEW
113 RTN
114 LBL 01
115 '__■-----
116 ARCL 05
117 RTN
118 LBL 02
119 '__
120 ARCL 05
121 '⊢■----
122 ARCL 05
123 RTN
124 LBL 03
125 '__
126 ARCL 05
127 '⊢-■---
128 ARCL 05
129 RTN
130 LBL 04
131 '__
132 ARCL 05
133 '⊢--■--
134 ARCL 05
135 RTN
136 LBL 05
137 '__
138 ARCL 05
139 '⊢---■-
140 ARCL 05
141 RTN
```

```
142 LBL o6
143 '__
144 ARCL o5
145 '├----*
146 ARCL o5
147 RTN
148 LBL o7
149 '__
15o ARCL o5
151 '├-----*
152 RTN
153 LBL 2o
154 SF o2
155 '∠
156 ASTO o5
157 '_R
158 ASTO o8
159 RTN
16o LBL 21
161 '↘
162 ASTO o5
163 '_L
164 ASTO o8
165 RTN
166 LBL 'RA
167 RCL o4
168 PI
169 +
17o X↑2
171 FRC
172 STO o4
173 *
174 INT
175 RTN
176 LBL 'GA
177 'I
178 ASTO o5
179 END
```

Das Programm wird mit XEQ'AB gestartet.Es erscheint V?. Es wird die Geschwindigkeit in m/s eingegeben,R/S.
Die Geschwindigkeit liegt nach meiner Erfahrung am Besten zwischen 3o und 5o m/s. Liegt sie darüber läßt sich der Wagen kaum längere Zeit auf der "Straße" halten.
Liegt sie darunter,so braucht man in den Kurven praktisch nicht mehr zu lenken.
Es gibt drei unterschiedliche Anzeigen:
1.Geradeaus: I--*--I
2.Rechtskurve: ∠--*--∠
3.Linkskurve : ↘--*--↘
Kurven werden zusätzlich durch einen Ton und die Anzeige Kurve R (L) angezeigt.
Wird nicht gelenkt so wandert der Wagen allmählich an den Kurvenaussenrand bis schließlich die Anzeige UNFALL das Fahrerischekönnen nachweist.
Gelenkt wird mit 1-4 für Linkskurve,4 am geringsten,1am stärksten,und 6-9 für Rechtskurve,6 am geringsten,9 am stärksten.5 bedeutet Geradeaus, dies braucht aber nicht eingegeben zu werden,da der Rechner immer automatisch geradeaus fährt,wenn ihm nichts anderes eingegeben wird.
Die Eingabe kann nur erfolgen wenn Flag oo gesetzt ist.
Dies sollte zügig vonstatten gehen,da man nur ca. 2 Sekunden Zeit hat.Wem dies zu kurz ist,der kann hinter Schritt 6o eine weitere PSE einschieben.
Der Rekord mit diesem Spiel waren einmal 1oooom ohne Unfall.Dann muß man allerdings schon eine Menge Zeit mitbringen.
Ich hoffe,daß die Spiele Euch Spaß machen.

Happy Programming

```
 01◆LBL "MAGIC"
 02◆LBL 06
FIX 0  CF 29  CF 01
SF 12  ADV
"0<MAGIC X<1?"  AVIEW
STOP  STO 00  ADV
"MAGISCHE"  PRA
"     ZAHLEN"  PRA  ADV
CF 12  1.005  STO 12  1
STO 13  STO 14

 24◆LBL 00
1.01  STO 11

 28◆LBL 01
RCL 00  9821  *  .211327
+  FRC  STO 00  10  *
INT  X=0?  GTO 01
STO IND 11  ARCL X  "├ "
ISG 11  GTO 01  GTO 02

 47◆LBL 04
ADV  "SUCHEN SIE IHRE"
PRA  "SCHLUESSELZAHL"
PRA  "X AN DER "  PRA
"Y. STELLE"  PRA  FS? 01
GTO 07  "WENN SIE "
"├FERTIG SIND, "  PRA
"DRUECKEN SIE"  PRA
"RUN"  PRA  STOP

 67◆LBL 07
ADV  "IHRE ZAHL IST"
PRA  "WAHRSCHEINLICH"
PRA  "DIE "  ARCL IND 13
"├ AN "  ARCL 14
"├. STELLE."  PRA
FS? 01  GTO 06  1
ST- 13  ST- 14  SF 01
ADV  "NOCH EINMAL"  PRA
"MIT ANDERER"  PRA
"SCHLUESSELZAHL."  PRA
GTO 04

 93◆LBL 02
PRA

 95◆LBL 03
RCL IND 13  ST+ 13
ST+ 14  10  RCL 13  X<Y?
GTO 03  ISG 12  GTO 05
RCL Z  ST- 13  ST- 14
GTO 04

109◆LBL 05
10  ST- 13  GTO 00
.END.
```

Hallo Fans !

Hier eine Verbesserung des in prisma November 198o veröffentlichten Programmes von Robert Klauo.

Dies Prgr. benötigt lediglich SIZE o15, ist dafür um ca. 5o % länger und bietet noch zusätzlich eine Spielmöglichkeit !
Es rät die Zahl des Spielers, bevor dieser zählt. Die Wahrscheinlichkeit für eine wahre Vorhersage ist wie bei der ersten Zahl 5/6. Ein Zeitschriften-Auszug über das Verfahren und das Prinzip ist in der Clubbibliothek erhältlich.
Zur Bedienung: XEQ SIZE o15 Prgr. eingeben (Karte in Magnetkartenbibl.) XEQ"MAGIC" Zahl zwischen 0 und 1 eingeben, R/S R/S --wenn gefordert

Nachfolgend noch der Probeausdruck mit der Zahl .123456

Tschüß und Happy Programming   Euer Gerd

Gerhard Schönemann, Kirchfeld 11, 3584 Zwesten

```
0<MAGIC X<1?

MAGISCHE
      ZAHLEN

 6 8 9 1 7 2 5 8 6 3
 6 4 9 6 8 3 9 3 4 4
 9 2 1 8 5 2 6 2 5 2
 2 5 1 2 7 3 8 8 2 7
 7 2 1 3 2 7 8 3 2 3

SUCHEN SIE IHRE
SCHLUESSELZAHL
X AN DER
Y. STELLE
WENN SIE FERTIG SIND,
DRUECKEN SIE
RUN

IHRE ZAHL IST
WAHRSCHEINLICH
DIE 8 AN 47. STELLE.

NOCH EINMAL
MIT ANDERER
SCHLUESSELZAHL.

SUCHEN SIE IHRE
SCHLUESSELZAHL
X AN DER
Y. STELLE

IHRE ZAHL IST
WAHRSCHEINLICH
DIE 7 AN 46. STELLE.

0<MAGIC X<1?
```

Andreas Schultze (490)
Erikaweg 5
2111 Handeloh-Höckel

SIZE 007
Start mit XEQ "ZAHLEN"
F 22

Programm "ZAHLEN"

Dieses Programm erzeugt eine Zufallszahl zwischen 1 und einer Zahl, die mit "HOECHSTZAHL?" abgefragt wird. Wenn der Rechner nach "ZAHL" fragt, wird geraten. Je nachdem, ob die geratene Zahl größer oder kleiner als die Zufallszahl ist, antwortet der Rechner mit "ZU GROSS" oder "ZU KLEIN". Wenn man richtig geantwortet hat, spielt der Rechner eine Melodie, zeigt "RICHTIG" und die Anzahl der Versuche an und fängt von Vorne an. Wenn man zu spät antwortet, reagiert der HP 41-C mit "ZU SPAET"

```
ØØ1 LBL "ZAHLEN"
ØØ2 CLRG
ØØ3 "HOECHSTZAHL?"
ØØ4 PROMPT
ØØ5 STO Ø6
ØØ6 LBL Ø1
ØØ7 RCL ØØ
ØØ8 9821
ØØ9 *
Ø1Ø ,211327
Ø11 +
Ø12 FRC
Ø13 STO ØØ
Ø14 RCL Ø6
Ø15 *
Ø16 INT
Ø17 1
Ø18 +
Ø19 STO Ø1
Ø2Ø LBL Ø2
Ø21 1
Ø22 ST+ Ø4
Ø23 TONE 7
Ø24 "ZAHL"
Ø25 PSE
Ø26 PSE
Ø27 FC?C 22
Ø28 GTO "ZU"
Ø29 STO Ø2
Ø3Ø RCL Ø1
Ø31 X>Y?
Ø32 GTO "ZUKL"
Ø33 RCL Ø2
Ø34 RCL Ø1
Ø35 X<Y?
Ø36 GTO "ZUGR"
Ø37 GTO "GLEICH"
Ø38 LBL "ZUKL"
Ø39 TONE 8
Ø4Ø "ZU KLEIN"
Ø41 AVIEW
Ø42 PSE
Ø43 GTO Ø2
Ø44 LBL "GLEICH"
Ø45 TONE 9
Ø46 TONE 9
Ø47 TONE 8
Ø48 TONE 8
Ø49 TONE 9
Ø5Ø TONE 9
Ø51 X<>Y
Ø52 X<>Y
Ø53 X<>Y
Ø54 X<>Y
Ø55 X<>Y
Ø56 X<>Y
Ø57 X<>Y
Ø58 X<>Y
Ø59 X<>Y
Ø6Ø X<>Y
Ø61 TONE 9
Ø62 TONE 9
Ø63 TONE 8
Ø64 TONE 8
Ø65 TONE 9
Ø66 TONE 9
Ø67 TONE 9
Ø68 "RICHTIG"
Ø69 AVIEW
Ø7Ø PSE
Ø71 CLA
Ø72 ARCL Ø4
Ø73 "├ VERSUCHE"
Ø74 AVIEW
Ø75 PSE
Ø76 Ø
Ø77 STO Ø4
Ø78 GTO Ø1
Ø79 LBL "ZU"
Ø8Ø TONE Ø
Ø81 "ZU SPAET"
Ø82 AVIEW
Ø83 PSE
Ø84 GTO Ø2
Ø85 END
```

```
ØØ1 LBL "SILBEN"
ØØ2 AON
ØØ3 CF Ø1
ØØ4 CF Ø2
ØØ5 CF Ø3
ØØ6 CF Ø4
ØØ7 CF Ø5
ØØ8 CF Ø6
ØØ9 CLA
Ø1Ø ASTO Ø1
Ø11 ASTO Ø2
Ø12 ASTO Ø3
Ø13 ASTO Ø4
Ø14 ASTO Ø5
Ø15 ASTO Ø6
Ø16 "1. SILBE"
Ø17 PROMPT
Ø18 FC?C 23
Ø19 GTO 99
Ø2Ø ASTO Ø1
Ø21 "2. SILBE"
Ø22 PROMPT
Ø23 FC?C 23
Ø24 GTO 99
Ø25 ASTO Ø2
Ø26 "3. SILBE"
Ø27 PROMPT
Ø28 FC?C 23
Ø3Ø GTO 99
Ø31 ASTO Ø3
Ø32 "4. SILBE"
Ø33 PROMPT
Ø34 FC?C 23
Ø35 GTO 99
Ø36 ASTO Ø4
```

Programm "SILBEN"

Dieses Programm kann Worte "verdrehen". Nach Eingabe eines bis zu 6-silbigen Wortes vertauscht es die Silben. Bei einem 4-silbigen Wort z.B. wird auf die Frage "5. SILBE" einfach R/S gedrückt.

SIZE 021
Start mit XEQ "SILBEN"
F Ø1
F Ø2
F Ø3
F Ø4
F Ø5
F Ø6
F 23

```
Ø37 "5. SILBE"
Ø38 PROMPT
Ø39 FC?C 23
Ø4Ø GTO 99
Ø41 ASTO Ø5
Ø42 "6. SILBE"
Ø43 PROMPT
Ø44 FS? 23
Ø45 ASTO Ø6
Ø46 LBL 99
Ø47 " MOMENTCHEN"
Ø48 AVIEW
Ø49 LBL Ø1
Ø5Ø FC? Ø1
Ø51 GTO Ø3
Ø52 FC? Ø2
Ø53 GTO Ø3
Ø54 FC? Ø3
Ø55 GTO Ø3
Ø56 FC? Ø4
Ø57 GTO Ø3
Ø58 FC? Ø5
Ø59 GTO Ø3
Ø6Ø FC? Ø6
Ø61 GTO Ø3
Ø62 GTO "ANZ"
Ø63 LBL Ø3
Ø64 XEQ Ø2
Ø65 XEQ IND X
Ø66 GTO Ø1
Ø67 LBL Ø2
Ø68 RCL ØØ
Ø69 9821
Ø7Ø *
Ø71 ,211327
Ø72 +
Ø73 FRC
Ø74 STO ØØ
Ø75 6
Ø76 *
Ø77 INT
Ø78 11
Ø79 +
Ø8Ø RTN
Ø81 LBL 11
Ø82 FS? Ø1
Ø83 GTO Ø1
Ø84 SF Ø1
Ø85 1
Ø86 ST+ Ø7
Ø87 ARCL IND Ø7
Ø88 ASTO 11
Ø89 RTN
Ø9Ø LBL 12
Ø91 FS? Ø2
Ø92 GTO Ø1
Ø93 SF Ø2
Ø94 1
Ø95 ST+ Ø7
Ø96 ARCL IND Ø7
Ø97 ASTO 12
Ø98 RTN
Ø99 LBL 13
1ØØ FS? Ø3
1Ø1 GTO Ø1
1Ø2 SF Ø3
1Ø3 1
1Ø4 ST+ Ø7
1Ø5 ARCL IND Ø7
1Ø6 ASTO 13
1Ø7 RTN
1Ø8 LBL 14
1Ø9 FS? Ø4
11Ø GTO Ø1
111 SF Ø4
112 1
113 ST+ Ø7
114 ARCL IND Ø7
115 ASTO 14
116 RTN
117 LBL 15
118 FS? Ø5
119 GTO Ø1
12Ø SF Ø5
121 1
122 ST+ Ø7
123 ARCL IND Ø7
124 ASTO 15
125 RTN
126 LBL 16
127 FS? Ø6
128 GTO Ø1
129 SF Ø6
13Ø 1
131 ST+ Ø7
132 ARCL IND Ø7
133 ASTO 16
134 RTN
135 LBL "ANZ"
136 ARCL 11
137 ARCL 12
138 ARCL 13
139 ARCL 14
14Ø ARCL 15
141 ARCL 16
142 PROMPT
143 GTO "SILBEN"
144 END
```

Programm "MUS"

Anhand des Listings dürfte die Funktion klar sei.

SIZE 001
Start mit XEQ "MUS"

```
01 LBL "MUS"
02 RCL ØØ
03 9821
04 *
05 ,211327
06 +
07 FRC
08 STO ØØ
09 10
10 *
11 INT
12 TONE IND Y
13 TONE IND X
14 TONE IND X
15 TONE IND T
16 TONE IND Y
17 TONE IND T
18 TONE IND Z
19 TONE IND Z
20 TONE IND X
21 TONE IND Y
22 TONE IND T
23 TONE IND X
24 GTO "MUS"
25 END
```

Sollte diese Seite schwer leserlich sein, so liegt das an dem sehr schlecht gestaltetem Original dieser Prgr., das ich von fünf auf jetzt 2 Seiten zusammengeschnitten habe. Demnächst werde ich Originale, die derart verschwenderisch mit dem wertvollen Platz in unserem prisma umgehen wollen, zurücksenden ! Oliver (1)

4. Liste zum KA-Programm

Das KA-Programm fragt bei jedem Durchgang nach 2 Bytes. Bei 1-Byte-Befehlen muß immer ein 2. Byte eingegeben werden, dessen Bedeutung aus der PPC-Tabelle in Prisma 9/80 nicht zu ersehen ist.

1. Byte unter 64

2.B./Anzeige sol. T. gedr./Taste losgelassen/Bemerkungen

2.B.	Anzeige sol. T. gedr.	Taste losgelassen	Bemerkungen
ØØ	CAT_	CAT_	
Ø1	Pc _	Keine Änderung	??
Ø2	DEL__	DEL__	
Ø3	COPY__	COPY__	
Ø4	CLP_	CLP_	
Ø5	Augenblickl.Z.	Programm läuft	Funktion R/S
Ø6	SIZE__	SIZE__	
Ø7	1Z.zurück	Keine Änderung	Funktion BST
Ø8	Augenblickl.Z.	Keine Änderung	Funktion SST
Ø9	ON	Keine Änderung	
1Ø	PACK	PACKING	
11		löscht augenblickliche Programmzeile	
12		schaltet PRGM-Modus ein und aus	
13	2__	2__	bleibt stehen
14		schaltet SHIFT ein und aus	
15	ASN_	ASN_	
16		Ø_	??
17	$T+N IA ___	$T+N IA ___	??
18			??
19	µ __	3 __	??
2Ø	DCABEE a	4	??
21	X	5	??
22	H LD	6	??
23	A	7	??
24	2 __	2 __	??
25	OD __	9	??
26	-	Ø (Schiebt X nach Y)	??
27	H	H	??
28	2__	2__	??
29	[illegible] AHHH	NONEXISTENT	??
3Ø	[illegible]NQ	NONEXISTENT	??
31	2	2	??
32	??		
33	A	A	??
34	IE[illegible]	[illegible]B	??
35	C __	C __	??

```
36                                           ??
37  ⌐bDCABP? ¯a      Keine Änderung          ??
38                                           ??
40   ¯2 __            ¯2                     ??
41 \Y                Keine¯Änderung          ??
42  ⌐bDCAB[]¯a       Keine Änderung          ??
43  OD               Keine Änderung          ??
44  A                A                       ??
45  dS7 µ            Keine Änderung          ??
46  X≤=Y?            Keine Änderung      ≠Befehl X≤=Y? !
47  µ                Keine Änderung          ??
48  ASCI__           ASCI__                  ??
49  9                9                       ??
50  REG¯             REG¯                    ??
51  ???????????????????????????????????????????
52  STX              Keine Änderung          ??
53    2 __             2 __                  ??
54    8 ¯              8 ¯                   ??
55  ???????????????????????????????????????????
56  OD               Keine Änderung          ??
57  0                Keine Änderung          ??
58  +                Keine Änderung          ??
59  >SF_             >SF                     ??
60      ¯bringt im PRGM-Modus die Anzeige zum Flacker
61   _                _                      ??
62   _                _                      ??
63   2                2                      ??
?!?!?!?!?!?!?!?!?!?!?!?!?!?!?!?!?!?!?!?!?!?!?!
```

Wer hilft mir, die Fragezeichen zu beseitigen ??

Ich hoffe, daß Ihr die Liste und die Programme verwenden könnt und verbleibe mit

Happy Programming

Euer Andreas

Sollte sich ein Mitglied für Programme auf dem Gebiet Umformtechnik (Warm- und Kaltwalzen, Schmieden, Drahtziehen o.ä.) interessieren, so helfe ich gerne weiter. Christian Herr (277), Rosstr. 63, 5100 Aachen, Tel. 0241/29022

P R O J E K T L E I T E R   P H Y S I K

Alles, was mit Physik zu tun hat, egal ob Software oder Berichte, sendet bitte an unseren neuen Projektleiter PHYSIK: Olaf Gursch (120), Grüntenstr. 29 b, 1000 Berlin 42

In der Januar-Ausgebe 1982 von prisma (Auflage 3000) sollen alle aktiven Mitglieder (Projektleiter, Ortsgruppenleiter etc.) mit Bild, Anschrift usw. vorgestellt werden. Daher: Wer zu diesem Kreis zählt, sende bitte Paßbild und Adresse/Telefon für 1982 bis allerspätestens Weihnachten an Oliver (1). Das Paßbild muß schwarz-weiß sein !

Liebe Mitglieder,

spielt jemand von Euch Bowling? Bitte schön, dann könnt Ihr mein Programm benutzen, aber auf mehr als 300 Pin pro Spiel kommt Ihr damit auch nicht, denn es arbeitet gemäß den gültigen Regeln.

Es übernimmt das Schreiben und Rechnen für einen Spieler und ermöglicht die Fortschreibung der Ergebnisse. Gestartet wird es mit XEQ "BOW".

Das Programm belegt 92 Register, für die Daten werden außerdem noch 13 Register gebraucht. Also, ein Memory-Modul ist Pflicht. Sollte jemand keinen Kartenleser haben, sind die Schritte 3 bis 8 und 134 bis 144 bei der Eingabe zu überspringen.

Kurz nach dem Start fordert es die Daten der vergangenen Spieltage an. Falls Ihr es zum erstenmal benutzt, bitte dreimal R/S drücken. Es ist an dieser Stelle aber auch möglich, Vorträge manuell abzuspeichern. Dazu bitte zweimal R/S drücken, die Daten eingeben und noch einmal R/S drücken.
Folgende Register stehen zur Verfügung:

```
Register 07 = Anzahl der bisher gemachten Spiele
         08 = Gesamt-Pin aus diesen Spielen
         09 = Höchstes Einzelspiel
```

Nach der Anzeige des aktuellen Spiels stoppt das Programm mit dem Text:

```
1 * 1. WURF
```

Hier müssen die Ergebnisse eingegeben werden. Dafür gibt es folgende Möglichkeiten:

```
0 Pin  =            R/S
1 Pin  = Ziffer und R/S
.                   .
.                   .
9 Pin  = Ziffer und R/S
Spare  =            "B"
Strike =            "C"
```

Die Flags im Display haben folgende Bedeutung:

```
Flag 00 = Spare wurde geworfen
     01 = 1 Strike wurde geworfen
     02 = 2. Strike in Folge wurde geworfen
     03 = 3. oder mehr Strike in Folge wurde geworfen
     04 = 10. Frame ist erreicht
```

Die Anzeige der Frame-Summen ist von den erzielten Ergebnissen abhängig. Nach einem Spare oder einem Strike muß erst der nächste oder die beiden nächsten Wurf erfolgen.

Nach dem 10. Frame wird die Endsumme angezeigt. Nun habt Ihr drei Möglichkeiten:

Neues Spiel = "A"
Werte anzeigen = "D"
Werte abspeichern = "E"

R/S ist an dieser Stelle unnötig, wenn Ihr eine der drei Tasten während des Pause-Befehls drückt.

Mit der Taste "D" werden folgende Werte angezeigt:

Gesamt 1 = Summe der Pin aus den aktuellen Spielen
Gesamt 2 = Summe der Pin aus allen Spielen
Schnitt 1 = Durchschnitt aus den aktuellen Spielen
Schnitt 2 = Durchschnitt aus allen Spielen
Spiele = Anzahl aller Spiele
Rekord = Höchstes Einzelspiel

Mit der Taste "E" werden die neuen Vorträge auf Karte abgespeichert. Flag 14 ist dabei gesetzt.

Unzulässige Eingaben werden von dem Programm nicht erkannt.

Bemerkungen: Der USER-Modus ist programmgesteuert
TONE 0 = Ausgaben
TONE 9 = Eingaben

Für Mitteilungen über Programm-Modifikationen bin ich dankbar.

Ansonsten: GUT HOLZ! und viele 300er Spiele

Mit freundlichen Grüssen

Peter

Peter Kark
(467)
Gerlachstrasse 33
6230 Frankfurt 80

Preise !

Aufgrund des auf 2,6o DM angestiegenen Dollarkurses mußte HP die Preise anziehen. Mitglieder bekommen bei Angabe der Mitgliedsnummer 2o bis 24 % Rabatt bei Firma Hansen, z. Hd. Herrn Milles, Bachstr. 1-3, 53oo Bonn1, Tel. o228/652223. Der gewährte Prozentsatz ist abhängig vom Lagerbestand. Beim Quad-RAM treten derzeit wieder Lieferschwierigkeiten auf, da HP mit der Produktion nicht nachkommt.
Hier die Preise (Stand heute, 21-o8-81):

HP-34c 46o,--	SD Memory und 4k-ROM		99,--	
HP-41c 75o,--	QD Memory		313,5o	
HP-41cv 975,--	8k-ROM		148,5o	
HP-67 1185,--	Lösungsbücher		4o,62	
HP-97A 217o,--	Magnetkartenleser	HP-821o4A		645,--
	Thermodrucker	HP-82143A		1155,--
	Opt. Lesestift	HP-82153A		375,--

(unverbindliche Preisempfehlung, für Clubmitglieder 2o bis 24 % Rabatt)

Oliver (1)

Listing für "BOWLING": (Die Magnetkarten dazu kann man sich in der Magnetkarten-bibliothek beschriften lassen !!)

```
01•LBL "BOW"
CLRG "DATENKARTE ?"
TONE 9 TONE 9 AVIEW
7.009 RDTAX

09•LBL A
CLA FIX 0 CF 26 CF 29
.00901 STO 00 0
STO 03 STO 04 STO 05
1 ST+ 06 ST+ 07
"PINS" ASTO 11 CLA
ARCL 06 "⊢. SPIEL"
BEEP XEQ 07 CLX
XEQ 00

32•LBL B
CF 27 "S P A R E"
XEQ 07 0 STO 01 10
STO 03 FS?C 01 XEQ 11
FS?C 06 XEQ 05 FS? 04
SF 05 FS? 05 XEQ 03
SF 00 XEQ 00

50•LBL C
CF 27 "S T R I K E"
XEQ 07 10 FS? 01 20
FS? 06 20 FS? 02 30
FS? 07 30 STO 04
FS? 00 XEQ 06 FS? 00
0 FS?C 00 STO 03
FS? 07 SF 08 FS? 08
XEQ 05 FS? 06 SF 07
FS? 07 XEQ 02 FS?C 05
XEQ 05 FS? 02 SF 03
FS? 01 SF 02 FS? 04
SF 06 FS? 06 XEQ 00
SF 01 XEQ 00

90•LBL D
"GESAMT" ASTO 12 "⊢ 1"
XEQ 07 ARCL 10 ARCL 11
XEQ 07 ARCL 12 "⊢ 2"
XEQ 07 ARCL 08 ARCL 11
XEQ 07 "SCHNIT"
ASTO 12 "⊢T 1" XEQ 07
RCL 10 RCL 06 / FIX 2
ARCL X ARCL 11 XEQ 07
ARCL 12 "⊢T 2" XEQ 07
RCL 08 RCL 07 /
ARCL X ARCL 11 XEQ 07
FIX 0 ARCL 07
"⊢ SPIELE" XEQ 07
"REKORD" XEQ 07
ARCL 09 ARCL 11 XEQ 07
XEQ D

134•LBL E
CF 27 "DATENKARTE ?"
TONE 9 TONE 9 AVIEW
SF 14 7.009 WDTAX CLX
OFF

145•LBL 00
FS?C 03 XEQ 11 FS? 06
XEQ 01 ISG 00 XEQ 01
TONE 0 TONE 0 TONE 0
SF 04

156•LBL 01
ARCL 00 FC? 06
"⊢ • 1. WURF" FS? 06
"⊢ • 2. WURF" XEQ 08
STO 01 FS?C 00 XEQ 06
0 STO 03

168•LBL 02
FS?C 02 XEQ 11

171•LBL 03
ARCL 00 FS? 06 CF 01
FS? 06 "⊢ • 3. WURF"
FS? 06 XEQ 04 FC? 05
"⊢ • 2. WURF" FS? 05
"⊢ • 3. WURF"

183•LBL 04
XEQ 08 STO 02 FS?C 01
XEQ 11

188•LBL 05
XEQ 06 0 STO 01
STO 02 CF 05 CF 06
CF 07 CF 08 FS?C 04
XEQ 09 XEQ 00

200•LBL 06
TONE 0 TONE 0 RCL 05
RCL 01 + RCL 02 +
RCL 03 + RCL 04 +
STO 05 "SUMME" XEQ 07
ARCL 05 ARCL 11 XEQ 07
RTN

219•LBL 07
AVIEW TONE 0 PSE CLA
RTN

225•LBL 08
TONE 9 CLX SF 27
PROMPT CF 27 CLA RTN

233•LBL 09
RCL 05 ST+ 08 ST+ 10
RCL 09 RCL 05 X>Y?
X<> 09

241•LBL 10
"ENDSUMME" XEQ 07
ARCL 05 ARCL 11 XEQ 07
SF 27 XEQ 10

249•LBL 11
XEQ 06 10 ST- 04
.END.
```

```
01•LBL "SKAT"
FIX 0 CF 29 1
"SPIELERZAHL?" PROMPT
STO 15 3 X=Y? SF 03
RDN 1 E3 / + STO 14
STO 00 LASTX + RCL 15
+ STO 11 1,001
FS?C 03 ST+ 11 RCL 11
STO 12

27•LBL 00
AON "NAME" PROMPT
ASTO IND 00 AOFF
ISG 00 GTO 00

35•LBL 01
RCL 14 STO 00

38•LBL 02
RCL IND 00 STOP

41•LBL A
ST+ 05

43•LBL 03
ST+ 16 ISG 00 GTO 02
GTO 01
```

Otto Peters (...9o), Boschstr. 18 b, 2850 Bremerhaven

Hinweise zum Programm Skat.

Nach dem Start fragt der Rechner zunächst nach der Anzahl der Spieler.

Man gibt dann 3 bzw. 4 ein und drückt R/S. Damit wird die indir. Adreßroutine errechnet und gleichzeitig die Routine zur Berechnung des Endergebnisses festgelegt.

Dannwird nach den Namen der Spieler gefragt. Eingabe im bereitgestellten Alpha-Modus.
(Zu beachten, daß die Register nur sechs Buchstaben aufnehmen können,also ggf. kürzen)

Sind drei oder vier Namen eingegeben, so steht jetzt wieder der erste in der Anzeige. Gleichzeitig sind den Namen die Tasten A - C bzw. D zugeordnet.

Der in der Anzeige stehende Spieler ist Kartengeber.

Gewinn und Verlust werden (positiv odernegativ) über die Tasten A - C(D) dem jeweiligen Spieler angeschrieben und in der Anzeige erscheint der Name dessen, der nun geben muß.

Zwischenabfragen nach dem Kontenstand können beliebig oft über die Taste H abgerufen und ausgedruckt werden.

Zur Endabrechnung drückt man E. Damit wird der letzte Kontenstand und darunter die Ausrechnung ausgedruckt.

Generallöschung = Taste F

```
 48•LBL B
ST+ 06  GTO 03

 51•LBL C
ST+ 07  GTO 03

 54•LBL D
ST+ 08  GTO 03

 57•LBL H
RCL 14  STO 10  RCL 12
STO 11  CLA

 63•LBL 04
"├ "  ARCL IND 10
ISG 10  GTO 04  ACA  ADV

 70•LBL 07
4  RCL IND 11  X=0?
GTO 08  ABS  LOG  INT  -

 79•LBL 08
SKPCHR  RCL IND 11  ACX
ISG 11  GTO 07  ADV
FS? 00  GTO e  GTO 02

 89•LBL E
SF 00  GTO H

 92•LBL e
RCL 12  STO 11

 95•LBL 11
RCL 15  ST* IND 11
ISG 11  GTO 11  RCL 12
STO 11

102•LBL 10
RCL 16  ST- IND 11
ISG 11  GTO 10  RCL 12
STO 11  CF 00  GTO 07

111•LBL F
CLRG  END
B-HAVEN, 10061981
```

```
Zwischen   HANS   KARL   FRITZ
             86    116      -8
abfragen   HANS   KARL   FRITZ
            206    160     -44

END-       HANS   KARL   FRITZ
            206    160     -44
ABR.        296    150    -454

Zw    HANS   KARL   FRITZ   HEINZ
      -120     48      40      36

End   HANS   KARL   FRITZ   HEINZ
      -120     48      40      36
      -484    186     156     140
```

Liebe Mitglieder !

"Was ist denn das für eine seltsame Kopfzeile ?" mag sich manch einer fragen, wenn er das CCD e.V. liest. Diese drei Buchstaben stehen für 1oo1 Änderungen in unserem Club. Es begann mit dem Artikel "Ausblick 82" im letzten prisma und fand auf dem letzten Clubtreffen in Frankfurt (1o. und 11. Oktober) seinen vorläufigen Höhepunkt: CCD steht für Computerclub Deutschland, der am 11.1o. in Kronberg/Taunus gegründet wurde. Zum Zeitpunkt des Erscheinens dieses Infos wird der Verein wohl bereits in das Vereinsregister Königstein/Taunus eingetragen und als gemeinnützig anerkannt sein. Nachstehend findet Ihr Satzung und Gründungsprotokoll. Der CCD übernahm am 11.1o. alle Tätigkeiten des HPAC, der zum Jahresende erlischt. Die Mitglieder des HPAC werden mit Bezahlung des Beitrags für 1982 Mitglied im CCD !
Vorteil 1: Die Mitgliedsbeiträge, Spenden, Aufnahmegebühren, ... können von allen Mitgliedern steuerlich voll abgesetzt werden. Das eröffnet die Chance, daß sich finanzkräftige Mitglieder vielleicht doch einmal zu einer Spende hinreißen lassen könnten. In diesem Zusammenhang sei Werner Dworak (6o7) für eine Spende über 2oo,--DM gedankt !
Vorteil 2: Die persönl. Haftung der Mitglieder beschränkt sich auf den Mitgliedsbeitrag; für die aktiven Mitglieder wird zusätzlich eine Vereinshaftpflichtversicherung abgeschlossen. Dies wurde angesichts der immer größeren Geldmengen, die der Club umsetzt, notwendig. So befindet sich in der Finanzplanung für 1982 noch ein mehrere zigtausend Mark großes Finanzloch. Hierfür mag verständlicherweise kein Mitglied den Kopf hinhalten, auch wenn wir davon ausgehen, daß die für 1982 geplanten Aktivitäten noch mehr Geld einbringen werden....

Steuerliche oder sonstige Vorteile, wie bereits einmal erörtert, kann der Club z.Zt. aber nicht erwarten; die Eintragung dient vornehmlich den Mitgliedern ! Die Haftungsvorteile übersteigen inzwischen sogar schon die seinerzeit erörterten Nachteile. Trotzdem: Diese Überlegungen haben wir schon einmal angestellt, warum nun doch e.V. !??!
1: Damals waren wir unter 4oo Mitgliedern, heute fast 1ooo ! Damit werden erheblich größere Geldmengen umgesetzt, die Verwaltung per EDV wird nötig usw.
2: Das Image eines gemeinnützigen e.V. wird uns in unseren Bestreben, möglichst viel für den HP-41 und kommende Rechner zu veröffentlichen, durch Vertrauensgewinn bei Firmen (Werbeanzeigen auf letzter Umschlagseite, Spenden) und steuerliche Vorteile (Abzugsfähige Spenden) sehr unterstützen, indem es unser Finanzloch stopfen hilft.
3: prisma bleibt HP-41 spezifisch. Zusätzlich aber sollen (wenn auch etwas weniger umfangreich) eigene Infos für den HP-85 und den Osborne 1 erscheinen. Der HP-85 wurde bereits im Dezember 80 Info von prisma kurz vorgestellt und dürfte inzwischen weitgehend bekannt sein. Nicht so der OSBORNE 1 ! Daher findet sich in dieser Ausgabe ein ausführlicher Bericht über den Osborne1. Er ist zugleich Antwort auf meinen Personalcomputerbericht in prisma 196-81 und die beiden Apfel(II)-Besitzer, die diesen Bericht als "unqualifiziert" bezeichneten. Meine Frage nun: Welcher Rechner ist unqualifiziert, der Apple oder der Osborne ???
Da ich gerade bei Kritik angelangt bin: Der PPC - Amerikas größter HP-Club mit rund 25oo Mitgliedern, davon etwas mehr als die Hälfte HP-41 Besitzer - hat mich schwer enttäuscht: Seit der Vorsitzende - Richard Nelson - erfahren hat, daß ich Vorsitzender eines deutschen "Konkurrenzladens" bin, verweigert er mir das PPC-Journal. Freilich hinderte ihn das nicht daran, meine 34 Dollar zu kassieren ... den Beitrag für 1981 ! Ich werde natürlich meine Konsequenzen ziehen - wer wirft sein Geld schon gerne weg ? Dabei scheine ich nicht der einzige zu sein: Während aus unserem Club bis 1982 voraussichtlich unter 2% der Mitglieder wieder austreten werden, so werden es beim PPC schätzungsweise zwei Drittel der Mitglieder sein. Das hindert mich aber nicht daran, das PPC-Journal weiterhin zu lesen. So entdeckte ich zum Beispiel die Bauanleitung für das Kassetteninterface von Ulrich Jansen(183). Warum schenkst Du den Amis die Bauanleitung und verweigerst sie uns, Ulrich ? Ich hatte Dich doch sogar schon darauf angesprochen ! Aber auch andere Mitglieder produzierten enttäuschendes: Edmund Weitz hat gerade drei Seiten der Infos für seine Umfragebögen verbraucht, da tritt er aus dem Verein aus ! Oder Rolf Mach: Übernimmt die Synthetic-Abteilung um das Buch von Dr. Wickes zu empfehlen, anstatt irgendetwas selbständiges zu produzieren. Statt dessen hat er Material aus höchst zweifelhafter Quelle - oder besser: eindeutiger Quelle in zweifelhafter Weise - über CHIP zum Kauf angeboten, z.B. den WPRV-Knack, den er allerdings über einen "Strohmann" verkauft - 20,--DM !! Oder Load Bytes - das wir deshalb (auch deshalb) lieber in dieser Ausgabe veröffentlichen - bevor der Wucher Überhand nimmt.

Damit aber genug Kritik für diese Ausgabe, aber es sind Themen, die halt einmal gesagt wer den müssen. Zu diesen Kritikpunkten habe ich schon viele dutzend Briefe bekommen, bevor ic mich zu diesem - inzwischen wohl nötigen - Absatz durchgerungen habe - in Vertretung der größten Mehrheit hoffe ich ! Also: Schnell zu den erfreulichen. Themen: Was gibts neues in und um prisma ??

prisma 12-81 - 12-81 - 12-81

Der Leserbrief von Klaus-Werner und Niels (215-81) hat Resonanz gefunden: Niels Nöhren hat zwei sehr gute Berichte zur synthetischen Programmierung verfaßt, die die gewünschten Grur lagen bringen dürften. Niels entwickelt sich zum Star-Autor !
Von Olaf Gursch finden wir nicht nur ein Reisekostenprogramm, sondern auch die Auswertung der Clubwahl 1980/1. Vielen Dank für die erstklassige Arbeit, Olaf ! Und natürlich allen, die dabei gut abgeschnitten haben ! Die anderen schaffen es ja vielleicht im nächsten Jahr Und wir wissen endlich einmal genau, was die Vereinsmitglieder so interessiert ! Ansonster Clubbörse, viele Programme, Leserbriefe und vieles mehr ! Viel Spaß !

prisma 1-82 - 1-82 - 1-82

Ab dieser Ausgabe wird prisma von Grund auf erneuert. Sie wird in Auflage 3.ooo oder mehr gedruckt und auch Nichtmitgliedern als Ansichtsexemplar zugänglich sein. Alle aktiven Mitglieder senden dafür bitte umgehend noch ein Paßfoto (schw./weiß) ein !

prisma 2-82 - 2-82 - 2-82

Die erste Ausgabe im neuen Stil ! Nur für Mitglieder und hoffentlich inberessant und ergre fend von der ersten bis zur letzten Zeile. Mehr wird nicht verraten ! Evtl. werden 1-82 und 2-82 zusammen an die Mitglieder versandt !

C L U B B Ö R S E

Verk. 1 Mem.Modul mit Buch "Synth.Prgr." WC.Wickes für 1oo,--DM
Burkhard Karallus (546),Reichstr.5o,597o Plettenberg
Suche PPC-ROM
Elk-Detlev Golz (5o7) Rugierweg 8, 588o Lüdenscheid

Verkaufe 2 Mem.Module je 5o,--DM
Dieter Riethenauer, Am Pfädle 3, 7519 Gemmingen, o71313oo66 bü

Verkaufe 1 Mem.Modul VB 5o,--DM und CHIP 5-79 bis 9-81 nicht unter5o,--DM (außer 7-79). Erstelle auch gerne Druckerlistings gegen Rückporto und Programmbeschreibung
Eckhard Müller, Michael-Schnabrich-Str.9, 35oo Kassel o561492266

Verk. MÜHLE-Prgr. für 41. Keine vereinfachte Version ! Spiel-stark und schnell (weniger als 2min./Zug). Magnetkartenleser und QDM (oder cv) sind erforderlich. 12DM plus 7DM für 12 Karten
Wer verkauft oder kopiert mir alte KeyNotes bis Ende 8o ?
Michael Kirchner (421), Goethestr. 29, 32oo Hildesheim

Verk. HP41c 5oo,--DM, Drucker 7oo,--, Kartenleser 35o DM, Opt. Leser 2ooDM, QDM 25o DM, MemMod 4oDM,2Akkusätze 2o,--DM,1 Experimentierplatine 2o,DM 15 Rollen Thermop. 15DM 11o MagCards XXXXX 4oDM Ext.Bat.Kasten 1oDM Basiccompiler 4oDM, zusammen 2oooDM: S Schwarze(41), Hilbleetr. 34, 8ooo München 19 o89193649

Zu verkaufen: Thermorollen für 41-Drucker, doppelte Länge wie gewöhnlich (5om), klarer,blau-schwarzer Druck, zum Preis von 25m-Rollen (4 Rollen 11,2o incl. Porto,Verp.). Detlev Bock (2) Petrikirchstr. 36 34oo Göttingen o551 33221

Suche PPC-ROM
Hannes Wolf (479), Ottakringerstr. 64 / 9, A-117o Wien

Verkaufe HP-41c mit 1 SDMemory und 1 DDMemory, 600,--DM VHB
Oliver Rietschel (1), Postfach 373, 2420 Eutin 1

Clubtreffen

Immer mehr Clubtreffen werden in den verschiedensten Städten veranstaltet. Das erste fand in Frankfurt/M. statt. Dazu erhielt ich folgenden Bericht von Matthias Grabiak:

"Das erste Clubtreffen im Raum Frankfurt hat nun am ersten August stattgefunden; zwölf Teilnehmer hatten sich dazu eingefunden. Konkret ins Leben gerufen wurde dabei eine Art Einführungskurs für synthetische Programmierung, um für alle Teilnehmer etwa gleiche Voraussetzungen zu schaffen, während sich gleichzeitig eine andere Gruppe über fortgeschrittene synthetische Programmierung unterhielt. Neben der synth.Prgr. wurde zwar kein weiteres spezielles "Programm" beschlossen, aber, wie ich persönlich glaube, wird es bei den Treffen von selbst genügend Gesprächsstoff geben.
Stattfinden sollen die Treffen an jedem zweiten Samstag eines Monats, 15.oo Uhr. Ein Raum wird freundlicherweise von der katholischen Studentenverbindung bereitgestellt, sofern keine anderen Veranstaltungen stattfinden. Hier die Adresse: 6ooo Frankfurt/M., Sömmeringstr. 3. Die Sömmeringstr. ist eine Seitenstr. des Oederweges, der parallel zur Eschersheimer Landstraße verläuft." Matthias Grabiak

Leider waren die Frankfurter die einzigen, die mir bisher einen solchen Bericht sandten. Alle anderen Ortsgruppen mögen das bitte schnellstens nachholen !!! Hier noch ein paar Kontaktadressen:
Dortmund: Karsten Beierlein (93), Breierspfad 83, 46oo Dortmund - Wambel
Stuttgart: Günther Lehrke (239), Wiesenstr. 85, 7o24 Fildarstadt 1
Kassel: Detlev Bock (2), Petrikirchstr. 36, 3400 Göttingen - Weende
Berlin: Hagen Klemp (73), Berliner Str. 33, 1ooo Berlin 28 Oliver(1)

-+

Was sollen die Flags ?

Oft werde ich gefragt, was die versch. Flags für Bedeutungen haben. Nachstehend eine Tabelle die die mir bekannten Verwendungszwecke wiedergibt. Für Ergänzungen bin ich jederzeit dankbar ! Oliver(1)

Flag	Bedeutung
00 - 11	allgemeine Anwenderflags zur freien Verwendung
00 - 07	Anwendungszweck wird bekonntgegeben, sobald entsprechende Peripherie erschienen ist ! Zur Zeit keine besondere Funktion !
11	Prgr., die über das Tastenfeld oder durch Einlesen einer Magnetkarte in den Rechner gerufen werden, werden automatisch ausgeführt, wenn bei deren Speicherung Flag 11 gesetzt war.
12	Solange Flag 12 gesetzt ist, werden Zeichen auf der Ausgabeeinheit in Sperrschrift (doppelter Breite) ausgegeben.
13	Solange Flag 13 gesetzt ist, werden alle Buchstaben in Kleinschrift ausgegeben.
14	Flag 14 ermöglicht das Überschreiben überschreibungsgeschützter Prgr. und Daten, wenn es gesetzt ist. Nach Beendigung des Schreibvorganges wird es automatisch gelöscht.
15 ÷ 16	Anwendungszweck wird bekonntgegeben, sobald entsprechende Peripherie erschienen ist. Zur Zeit keine besondere Funktion !
17	
32	--
18 - 20	???
31	???
33 - 35	???

Die restlichen Flags sind im HP-41 Handbuch ausführlich vorgestellt.

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Adreßänderung: Larken Publications, der Verlag, bei dem das Buch von Dr. Wickes erhältlich ist, ist umgezogen. Die neue Adresse lautet: Larken Publications, 4517 NW Queens Avenue, Corvallis, Oregon 9733o. Bei Bestellung 11,5o US$ per Postanweisung im voraus überweisen. Ein deutsches Buch zur synth.Prgr. erscheint voraussichtlich Anfang 1982.

Bekanntlich kann man durch Löschen des Flags 55 die Rechengeschwindigkeit bei angeschlossenem Printer um ca 20-25 % erhöhen.
Die kleine PRinter-OFf-Routine löst die Aufgabe sehr schnell. Sie benötigt die Alpha-Reg, lässt aber den Stack (inkl. L) im ursprünglichen Zustand. Sie kann daher bedenkenlos irgendwo in einem Programm ausgeführt werden, sofern die Alpha-Reg nicht gerade benötigt werden. Bei der nächsten Ausführung wird Flag 55 wieder gesetzt.

Die Funktionsweise ist die folgende: Flag 55 ist das letzte Bit des 7. (und letzten) Byte von Reg d. Die Routine schiebt dieses letzte Byte an die erste Stelle im Reg d und verwandelt somit Flag 55 in Flag 07 (letztes Bit des ersten Bytes). Flag 07 wird dann ganz normal gelöscht bzw. gesetzt, anschliessend werden Stack und Reg d in den ursprünglichen Zustand zurückversetzt.

```
 01•LBL "PRO
F"
 02 X<> d
 03 X<> [
 04 "⊢••••••
"
 05 X<> [
 06 X<> d
 07 FC?C 07
 08 SF 07
 09 X<> d
 10 X<> [
 11 "⊢A"
 12 X<> \
 13 X<> d
 14 END
```

Alberto Lenz (551) - Nordstr. 23 - CH-8006 Zürich - 0041-1-362 72 93

P R O J E K L E I T E R   P L O T T E N

Michael Hartmann - 38o - Wendelsgrund 13 - 669o St. Wendel

P R O J E K T L E I T E R   H A R D W A R E

Werner Dworak - 6o7 - Allewind 51 - 79oo Ulm - Tel. o73o4/3274

Magnetkartenbeschriftung

Die ganzen Vorsichtsmaßnahmen mit Klarlack und Übermalen sind nach meiner Erfahrung völlig überflüssig, wenn man darauf achtet, daß man mit wasserfesten Filzschreibern oder auch mit sog. Folientusche beschriftet. Ich verwende einen Feinlinienstift und habe keinerlei Schwierigkeiten. Prof. Jochen Kux

Ich habe die gleichen Erfahrungen gemacht !! Oliver

Lösungsbuch Nr. 5 - Kryptographie

Das LöBu 5 umfasst 69 DINA4 Seiten, zehn Prgr. und ist komplett mit Flußdiagrammen/Struktogrammen dokumentiert. Der Preis wird um 5o,--DM liegen. Interessenten wenden sich bitte an Frank Altensen (117), Uhlandstr.9 , 6365 Rosbach 3 - Tel. o6oo7/14o9
Die enthaltenen Prgr. sind: 1) Verschiebechiffrierung 2) Invertierung 3) Datenschutz 4) Semiotische Analyse 5) Morsealphabet I + II 6) Numerik 7) Kryptanalyse 8) Portasche Chiffrierung

---

Hardwareverleih

Ursprünglich war vorgesehen, Hardware an Clubmitglieder zu verleihen. Drucker und Kartenleser wurden bereits mehrfach verliehen, was sich aber als äußerst uneffektiv herausstellte. Außerdem ist der Arbeitsaufwand durch ständiges Verschicken zu groß. Bei der jetzigen Zahl von Mitglieder gibt es daher keine andere Möglichkeit, als alle Verleihvorhaben ersatzlos zu streichen. Die PPC-ROMs des Clubs wurden an Clubmitglieder mit der Auflage, sie auch an andere Mitglieder zu verleihen, zum Selbstkostenpreis verkauft. Bisher (Ende Oktober) gibt es allerdings noch keine Anzeichen einer baldigen Auslieferung der ROMs. Oliver(
Wer trotzdem Peripherieeinheiten kennenlernen will, kann dies auf allen Clubtreffen !

Suche Mitglieder, die bereit sind, versch. Graphicen auf dem HP-Drucker zu erstellen.
Andreas Roemer, Eckenheimer Landstr. 3, 6ooo Frankfurt/M., Tel. o611/5942o1

MOmentan sieht es so aus, als wenn wir selbst unter Berücksichtigung der Vorjahresverluste zum Jahresende (zumindest fast) ohne Schulden dastehen. Damit wären die Finanzsorgen bis Anfang 1982 vorerst vom Tisch !!! Oliver(o1)

Protokoll der Vereinsgründung des Computerclubs Deutschland

1. Am 11.1o.1981 trafen sich in Kronberg, Limburger Straße 15, die auf beiliegender Anwesenheitsliste verzeichneten Personen zur Beschlußfassung über die Gründung eines Vereins zur Unterstützung von Wissenschaft und Forschung auf dem Gebiet Informatik und Computertechnologie.

   Herr Oliver Rietschel gab einen Überblick über seine bisherigen Aktivitäten auf diesem Gebiet und schlug gleichzeitig den Namen "Computerclub Deutschland" vor. Dieser Name wurde einstimmig angenommen.

2. Herr Rietschel schlug vor, Herrn Bock als weiteren Versammlungsleiter für die Versammlung und Herrn Hansmann als Protokollanten per Handzeichen zu wählen. Nachdem beide gewählt waren, schlug Herr Bock folgende Tagesordnung vor:

   1. Beratung und Beschluß über die Vereinssatzung
   2. Wahl des Vorstandes
   3. Festsetzung des Jahresbeitrages
   4. Übergang des HP Anwenderclubs Oliver Rietschel in den neuen Verein.

Zu 1.:
Die Satzung wurde nach Diskussion und verschiedenen Änderungen von Herrn Bock zur Abstimmung gestellt. Sie wurde per Handzeichen von allen Anwesenden angenommen. Der Beitritt der Anwesenden zu dem Verein wurde durch Unterschreiben der Satzung vollzogen.

Zu 2.:
Die Vorstandswahl wurde per Handaufheben durchgeführt und hatte folgendes Ergebnis:

		ja	nein	Enthaltung
1. Vorsitzender:	Herr Oliver Rietschel	6		1
2. Vorsitzender:	Herr Rolf Hansmann	6		1
Beisitzer:	Herr Detlev Bock	6		1
	Herr Andreas Marktscheffel	6		1

Die Gewählten erklärten, daß sie die Wahl annehmen.

Zu 3.:

Herr Rietschel schlug vor, den Jahresbeitrag für 1982 auf DM 6o,-- festzusetzen.
Dieser Vorschlag wurde einstimmig angenommen.

Zu 4.:

Der CCD soll rechtlicher Nachfolger des nicht eingetragenen "HP Anwenderclubs Oliver Rietschel" werden. Mit Bezahlung des Mitgliedsbeitrages für 1982 sind dieMitglieder dieses Clubs Mitglied im CCD.

Kronberg, den 12.1o.1981

(Rolf Hansmann)

Satzung des Computerclubs Deutschland - CCD

§ 1

Der Verein führt den Namen "CCD - Computerclub Deutschland" mit dem Zusatz "e.V." nach Eintragung und hat seinen Sitz in Kronberg. Er ist rechtlicher Nachfolger des "Hewlett-Packard Anwender Club".

§ 2

Der Computerclub verfolgt ausschließlich und unmittelbar gemeinnützige Zwecke im Sinne des Abschnitts "Steuerbegünstigte Zwecke" der Abgabenordnung. Zweck des Vereins ist die Förderung von Wissenschaft und Forschung. Der Satzungszweck wird verwirklicht insbesondere durch Durchführung wissenschaftlicher Veranstaltungen und Forschungsvorhaben und Herausgabe eines vereinseigenen Informationsblattes.

§ 3

Der Verein ist selbstlos tätig; er verfolgt nicht in erster Linie eigene wirtschaftliche Zwecke.

§ 4

Mittel des Vereins dürfen nur für die satzungsmäßigen Zwecke verwendet werden.

§ 5

Es darf keine Person durch Ausgaben, die dem Zweck der Körperschaft fremd sind, oder durch unverhältnismäßig hohe Vergütungen begünstigt werden.

§ 6

Die Aufnahme in den Verein erfolgt auf Antrag. Die Mitgliederzahl ist unbeschränkt.

§ 7

Die Mitgliedschaft endet durch Tod, Austrittserklärung oder Ausschluß. Der zum Ende eines jeden Kalenderjahres mögliche Austritt erfolgt durch formlose Erklärung mit einer Frist von vier Wochen zum Jahresende an den Vorsitzenden. Über den Ausschluß entscheidet die Mitgliederversammlung mit einer Mehrheit von 75 % nach freiem Ermessen.

§ 8

Über die Höhe der Geldbeiträge beschließt die Jahresversammlung der Mitglieder nach einfacher Mehrheit. Das gleiche gilt für Satzungsänderungen und -ergänzungen.

§ 9

Organe des Vereins sind der Vorstand und die Mitgliederversammlung.

§ 1o

Der Vorstand besteht aus dem 1. Vorsitzenden, dem 2. Vorsitzenden und zwei Beisitzern. Vorstand im Sinne des § 26 sind der 1. und der 2. Vorsitzende, die den Verein jeweils allein vertreten können.

§ 11

Die bis zum 1. November eines jeden Jahres stattfindende Mitgliederversammlung beschließt über die Höhe der Beiträge, die Entlastung des Vorstandes, die Wahl des Vorstandes und über Satzungsänderungen. Der Vorstand im Sinne des § 26 BGB wird jeweils auf die Dauer von drei Jahren gewählt. Eine Wiederwahl ist zulässig. Über die Wahl der übrigen Mitglieder des Vorstandes wird jährlich entschieden. Die Einberufung der Mitgliederversammlung erfolgt durch den Vorstand mit einer Frist von vier Wochen schriftlich unter Bekanntgabe der Tagesordnung in der Clubzeitschrift. Die ordnungsgemäß einberufene Mitgliederversammlung ist unabhängig von der Zahl der erschienenen Mitglieder beschlußfähig. Über die Mitgliederversammlung ist eine vom Vorsitzenden oder von einem von der Versammlung gewählten Protokollführer zu unterzeichnende Niederschrift aufzunehmen.

§ 12

Bei Auflösung oder Aufhebung des Vereins oder bei Wegfall seines bisherigen Zwekkes fällt das Vermögen des Vereins an eine Körperschaft des öffentlichen Rechts oder eine andere steuerbegünstigte Körperschaft zwecks Verwendung für die Förderung von Wissenschaft und Forschung.

Unterschriften der Gründungsmitglieder siehe Rückseite.

Olaf Gursc[illegible] (120)
Grüntenstra[illegible] 29 b
1000 Berlin [illegible], den 23.10.1981

Liebe Clubmitglieder!

Oliver fragte vor einiger Zeit bei mir an, ob ich den Wahlleiter bei der anstehenden Wahl machen wolle. Ich stimmte gern zu. In dieser Eigenschaft haben viele von Euch in letzter Zeit Ihre Stimmzettel an mich eingesandt. Ich weiß nicht,ob Oliver eine Statistik darüber führt, wieviel Mitglieder aktiv am Club mitarbeiten, aber bei dieser Art Aktivität hatte ich mit mehr Beteiligung gerechnet. Nach den Eingängen der Stimmzettel dürfte unser Club zur Zeit ca. 700 Mitglieder haben (?). Zurückgekommen sind aber nur 180 Stimmzettel, wenn man von einigen wenigen Austritten absieht, also nur ca. 25% der Mitglieder haben sich an der Wahl aktiv beteiligt. Eigentlich doch sehr sehr schade. Denen,die sich an der Wahl beteiligt haben,gilt der Dank um so mehr. Nun zur Wahl selbst:
Es wurden insgesamt 93 verschiedene Beiträge genannt. Es entwickelte sich eigentlich von Beginn an ein Kopf an Kopf Rennen zwischen vier Beiträgen. Mit wechselnden Führungen lag eigentlich bei jedem Zwischenstand ein anderer der vier an der Spitze. Der Sieger kristallisierte sich erst mit den letzten ausgewerteten Stimmzetteln heraus:

```
***************************************************************
* 1. Bar Codes und Graphik mit dem 41C: Winfried Maschke (413) *
*       Seite 249 bis 252 -81          46 Punkte               *
***************************************************************
```

2. Plotter : Fred Huber Seite 114 bis 117 -81 41 Punkte
3. Hindernisfahrt: Ralf Wirowski Seite 262 bis264-81 40 Punkte

Dem Sieger herzlichenGlückwunsch!
Die vollständige Ergebnisliste findet Ihr im Anhang. Bei gleicher Plazierung sind die Beiträge in alphabetischer Reihenfolge der Autoren aufgeführt.
Bei der Suche nach dem 'erfolgreichsten' PRISMA-Autor gingen die Punkte, die jeder Beitrag erhielt,entsprechend ein. Wie es sich zeigte, waren die gleichen vier Autoren vorn wie bei der ersten Wertung. Die Reihenfolge war aber doch beträchtlich verschieden von der ersten, wenn man bedenkt, daß der Sieger nicht unter den ersten Drei der ersten Wertung war:

```
*********************************************************
* 1. Klaus Werner Hoenow (11)   mit insgesamt 50 Punkten *
*********************************************************
```

2. Winfried Maschke (413)   mit insgesamt 48 Punkten
3. Fred Huber (186)   mit insgesamt 42 Punkten

Dem Sieger herzlichen Glückwunsch!
Die vollständige Liste mit der Angabe der einzelnen Programme und der Punktwertung hierfür ebenfalls im Anhang.
Abgerundet wurde der Stimmzettel mit einer Befragung nach den Schwerpunkten der Interessensgebiete. Die folgenden prozentualen Angaben beziehen sich natürlich auf die 180 ausgewerteten Einsendungen:

Hardware	102	56,6%	Synthetisches	109	60,6%
Spiele	82	45,6%	Programmierhilfen	71	39,4%
Elektrotechnik	40	22,2%	Steuer/Finanzen	33	18,3%
Statik	17	9,4%	Statistik	34	18,9%
Mathematik	97	53,9%			

Mehr als eine zusätzliche Nennung erhielten die Gebiete:
Amateurfunk,Astronomie,Hoch-Tiefbau,Vermessung, Physik.
Darüber hinaus wurden weitere 33 Schwerpunkte je einmal aufgeführt.
Ihr findet die Liste im Anhang.
Den Siegern mit Ihrem neuen ROM viel Spaß. Ich hoffe bei der nächsten Wahl auf mehr Beteiligung!

ERGEBNIS ZUR WAHL DES BESTEN BEITRAGES AUS PRISMA 1980 / 81

Nr.	Beitrag	Autor	Fundstelle	Stimmen
1.	**Bar Codes & Graphik**	**Maschke**	**249/252-81**	46
2.	**PLOTTER**	**F. Huber**	**114/117-81**	41
3.	**Hindernisfahrt**	**Wirowski**	**262/264-81**	40
4.	**Matrizenrechnung**	**Hoenow**	**9/23-81**	38
5.	**Monatslohnabrechnung**	**Berg**	**68/76-81**	22
6.	**Rechnerorganisation**	**Marktscheffel**	**147/151-81**	21
	Rechnerorganisation	**Rietschel**	**Juli 80**	21
8.	**Programmierbarer Byte Jumper**	**Mach**	**41-81**	19
	Doppel Hires Plotting	Stroinski	44/46-81	19
10.	Rechnerorganisation	Grabiak	W.Sonder Heft	16
11.	Primfaktorzerlegung	Hillebrandt	169/183-81	15
	PLOTTER	Klauc	November 80	15
13.	Inverses Mastermind	Davertzhofen	W.Sonder Heft	11
	ORDO	Meyer-Lindenberg	257/260-81	11 +2
	Balkendiagramm	Pochert	247/248-81	11
16.	Key Assignment Program	Bock	18/Juli-80	9
17.	Paket: Interpolation	Ehrhardt	77/84-81	8
	Programmierhilfen Anfäng	Hoenow	Diverse	8
	High Resolution Plotting	Rietschel(Schwarz)	24/September-80	8
	QTH-Kenner	Spiegel	5-81	8
	Das sind Bar Codes	Warmuth	40-81	8
22.	Kfz-Kosten	Nöhren	85-81	7
	Character Builder	Pfeifer	241-81	7
24.	Kalenderausdruck	de Arras	23/September-80	6
	Kegeln	Rietschel	187-81	6
	FORMEL 1	Weitz	22/Juli-80	6
27.	PBJ-Status	Klemp	235-81	5
	Phasen/Finsternisse	Lelarge	7/Juni-80	5
	KA Program	Marktscheffel	151-81	5
	Flugnavigation	Wienbeck	157-81	5
31.	Mastermind	Davertzhofen	W.Sonder Heft	4
	Bastelprogramm	Grabiak	W.Sonder Heft	4
	Differentialgleichungen	M.Huber	W.Sonder Heft	4
	Kalenderprogramm	Nöhren	Oktober-80	4
35.	Raten und Rechnen	Altensen	101-81	3
	Hex-Dez Wandler	Holm	128-81	3
	Bubble Sort	Kirchner	219-81	3
	Vierpole	Lütke Uphues	W.Sonder Heft	3
	Primfaktorzerlegung	Tetens	208-81	3
	Acht Damen Problem	Weiler	W.Sonder Heft	3
	Memory Spiel	Weitz	21/September-80	3
42.	CHIP-Preisausschreiben	Altensen	153-81	2
	FN	Born	W.Sonder Heft	2
	ACXR	Hoenow	8/Oktober-80	2
	Diskrete Fourier Transf.	Kropf	W.Sonder Heft	2
	Inverse Fourier Transf.	Kropf	W.Sonder Heft	2
	LOGIC	Lindner	292-81	2
	Bug 3 Simulator	Lütke Uphues		2
	Formänderung Kragarm	Lütke Uphues	W.Sonder Heft	2
	Rechengeschwindigkeit	Lütke Uphues	24-81	2
	BARA/BP	Maschke	251-81	2
	Bruchrechnung	Mirgel	26/September-80	2
	Newton-Verfahren	Müller	206-81	2
	Primfaktorzerlegung	Moeck	209-81	2
	Primfaktorzerlegung	Nöhren	229-81	2
	Synthetische Programmier.	Nöhren	223-81	2
	U-Boot Jagd	Rietschel	13/Juli-80	2
	Mantisse m der Zahl in X	Schu	122-81	2
	Ohmsche Dämpfung	Stroinski	141-81	2
	Triangle Solutions		135-81	2

Alle folgenden Beiträge haben jeweils einen Punkt [illegible] Ihrem Konto und sind somit gemeinsam auf Platz 61:

10 Pkte.Gauß Quadratur	Bock	Test Magnetkartenleser	Mirgel
Rechnermarkt	Weitz	2 Fkt. Plottprogramm	M.Huber
Hex-Dez Wandler	Beiersdorf	Schiffe versenken	Rietschel
Stern Dreieck	Meyer	∑	Siewert
Flag	Mach	Aussagenlogik	Schu
UEBE	Tetens	LIN	Born
Memory Saver	Beiersdorf	Batterie 41 C	Folprecht
Page Switching	Lütke Uphues	Magische Zahlen	Klauc
Tips und Tricks		Dreiecksberechnung	Stroinski
Widerstandsbestimmung	Berg	PLOT 1	Hoenow
LBLFT	Raabe	Erg. zu MULTI	Hoenow
Perepheriefunktionen	Nöhren	HF	Wirowski
Akustisches Interface	Grabiak	PBJ	Kropf
Additionsprogramm	Elsässer	Prüfziffernprogramm	F.Huber/Bruch
Trafo Berechnung	Henze	Root	Wolpers
Primzahlentest	Pfeifer	STAR TREK	Krumins
Magisches Quadrat	Meyer-Lindenberg		

Insgesamt 93 Beiträge genannt von 180 Einsendungen!

ERGEBNISLISTE NACH AUTOREN GEORDNET

1. Hoenow	Programmpaket:Matrizenrechnung	38	50 Punkte
	Programmierhilfen für Anfänger	8	
	ACXR	2	
	PLOT 1	1	
	Ergänzungen zu MULTI	1	
2. Maschke	Bar Codes und Graphik	46	48 Punkte
	BARA/BP	2	
3. F. Huber	PLOTTER	41	42 Punkte
	Prüfziffernprogramm	1	
4. Wirowski	Hindernisfahrt	40	41 Punkte
	HF	1	
5. Rietschel	Rechnerorganisation	21	38 Punkte
	High Resolution Plotting	8	
	Kegeln	6	
	U-Boot Jagd	2	
	Schiffe versenken	1	
6. Marktscheffel	Rechnerorganisation	21	26 Punkte
	KA Program	5	
7. Berg	Monatslohnabrechnung	22	23 Punkte
	Widerstandsbestimmung	1	
8. Stroinski	Doppel Hires Plotting	19	22 Punkte
	Ohmsche Dämpfung	2	
	Dreiecksberechnung	1	
9. Grabiak	Rechnerorganisation	16	21 Punkte
	Bastelprogramm	4	
	Akustisches Schalt Interface	1	
10. Mach	Programmierbarer Byte Jumper	19	19 Punkte
11. Klauc	PLOTTER	15	16 Punkte
	Magische Zahlen	1	
12. Davertzhofen	Inverses Mastermind	11	15 Punkte
	Mastermind	4	
12. Hillebrandt	Primfaktorzerlegung	15	15 Punkte
12. Nöhren	KfZ-Kosten	7	15 Punkte
	Kalenderprogramm	4	
	Primfaktorzerlegung	2	
	Synth. Programmierung	2	
	Perepheriefunktionen	1	

15. Meyer-Lindenberg	ORDO	11+2	12 Punkte 14*
	Magisches Quadrat	1	
16. Pochert	Balkendiagramm	11	11 Punkte
17. Bock	Key Assignment Program	9	10 Punkte
	10 Punkte Gauß Quadratur	1	
17. Lütke Uphues	Vierpole	3	10 Punkte
	Bug 3 Simulator	2	
	Formänderung Kragarm	2	
	Rechengeschwindigkeit	2	
	Page Switching	1	
19. Weitz	FORMEL 1	6	9 Punkte
	Memory Spiel	3	
20. Ehrhardt	Paket:Interpolation	8	8 Punkte
20. Pfeifer	Character Builder	7	8 Punkte
	Primzahlentest	1	
20. Spiegel	QTH-Kenner	8	8 Punkte
20. Warmuth	Das sind Bar Codes	8	8 Punkte
24. de Arras	Kalenderausdruck	6	6 Punkte
25. Altensen	Raten und Rechnen	3	5 Punkte
	CHIP-Preisausschreiben	2	
25. M.Huber	Differentialgleichungen	4	5 Punkte
	2 Funktionen Plott Prog.	1	
25. Klemp	PBJ-Status	5	5 Punkte
25. Kropf	Diskrete Fourier Transf.	2	5 Punkte
	Inverse Fourier Transf.	2	
	PBJ	1	
25. Lelarge	Phasen/Finsternisse	5	5 Punkte
25. Wienbeck	Flugnavigation	5	5 Punkte
31. Tetens	Primfaktorzerlegung	3	4 Punkte
	UEBE	1	
32. Born	FN	2	3 Punkte
	LIN	1	
32. Holm	Hex-Dez Wandler	3	3 Punkte
32. Kirchner	Bubble Sort	3	3 Punkte
32. Mirgel	Bruchrechnung	2	3 Punkte
	Test Magnetkartenleser	1	
32. Schu	Mantisse m der Zahl in X	2	3 Punkte
	Aussagenlogik	1	
32. Weiler	Acht Damen Problem	3	3 Punkte
38. Beiersdorf	Hex-Dez Wandler	1	2 Punkte
	Memory Saver	1	
38. Lindner	LOGIC	2	2 Punkte
38. Moeck	Primfaktorzerlegung	2	2 Punkte
38. Müller	Newton Verfahren	2	2 Punkte

Alle anderen bisher nicht genannten Autoren erscheinen nur einmal mit einem Beitrag in der Liste und haben nur je einen Punkt. Diese Autoren sind alle gemeinsam auf Platz 42.

SCHWERPUNKTTHEMEN: (jeweils nur ein einziges Mal genannt)

Fotografie	Rechnerorganisation	Navigation	Brettspie
Kaufm.Progr.	Technische Programme	Maschinenbau	Kinematik
Geodäsie	Getriebetechnik	Bildschirm	Geometrie
Hydraulik	Peropherieerweiterungen	Massenspeicher	Neuheiten
Bar Code Aufb.	Wahrscheinlichkeitsrechn.	Telefoncomputer	Druck
Graphik	Themen aus PPC	Zahlentheorie	Plottings
Barplott	Selbstbauanleitungen	Konstruktion	Antennen
Utility	Umwelttechnik	Rechnungswesen	Kalender
	Adapter / Compiler		

*Die beiden Punkte für das Programm ORDO von Meyer-Lindenberg die jeweils als 12 vormerkt sind resultieren aus einem Stimmzettel-Rücklauf erst am 20.10. wegen längerer Krankheit eines Clubmitgliedes. Die Reihenfolge bleibt jedoch erhalten. Lediglich die die punktgleichen Beiträge von Platz 13 haben jetzt Platz 14!

Einfache Zählschleifen unter Benutzung von ISG bezw. DSE sind im deutschen Handbuch (Seiten 164 ff.) beschrieben. Bei mehrfach ineinander geschachtelten Zählschleifen tauchen jedoch gelegentlich Schwierigkeiten auf.

Es sei folgende (triviale) Aufgabe gestellt: für verschiedene Wertepaare (X, Y) soll die Transformation in Polarkoordinaten (R, ∡) durchgeführt werden. Dabei sollen vor Beginn der Rechnung vom Benutzer der Bereich (XA bis XE) und die Schrittweite (DX) für X und entsprechend für Y auf Anforderung eingegeben werden (sämtliche Werte ≥ 0 und ganzzahlig).

In der äußeren Schleife wird sodann der gesamte Bereich der X-Werte und in der inneren Schleife zu jedem X-Wert der gesamte Bereich der Y-Werte durchlaufen. Das Struktogramm zu dieser Aufgabe sieht also folgendermaßen aus:

1.	Eingabe XA, XE, DX		
2.	Eingabe YA, YE, DY		
3.	Für X = XA bis XE, Schrittweite DX		
4.		Anzeige X	
5.		Für Y = YA bis YE, Schrittweite DY	
6.			Anzeige Y
7.			Berechnung R(X, Y) und ∡(X, Y)
8.			Anzeige R und ∡

In den einzelnen Zeilen soll sich folgendes abspielen (vergl. auch den Programmausdruck SCHL mit zugehörigen Nummern):

1. Auf die Frage "XA ↑ XE ↑ DX" werden diese Werte vom Benutzer, getrennt durch ENTER und abgeschlossen durch R/S, eingegeben. Das Unterprogramm LBL 00 berechnet daraus gleich den Schleifenzähler XA + XE/1000 + DX/100000 (siehe deutsches Handbuch Seite 163). Dieser Zähler wird nach R01 gespeichert.
2. Für den Y-Bereich gilt das unter 1. Gesagte. Der Schleifenzähler wird nach R00 gespeichert.
3. LBL 01, ISG 01 und GTO 01 bilden Anfang und Ende der äußeren Zählschleife, wobei durch ISG 01 jeweils nach dem Durchlaufen der Schleife der X-Wert um DX erhöht wird.
4. Hier wird der jeweils aktuelle X-Wert angezeigt, wobei ihm vorher durch INT die hinter dem Dezimalpunkt liegenden Teile von XE und DX abgeschnitten werden. Da der "bereinigte" X-Wert für die spätere Rechnung gebraucht wird, wird er nach R03 zwischengespeichert.
5. LBL 02, ISG 02 und GTO 02 bilden Anfang und Ende der inneren Zählschleife. Allerdings -und das ist der wesentliche Punkt dieser Betrachtung- ist für die innere Schleife (bei mehreren verschachtelten Schleifen für alle inneren) ein Vorbereitungsschritt erforderlich: der nach R00 gespeicherte Schleifenzähler muß vor Beginn der Schleife in das Register (hier R02) geladen werden, in dem anschließend die Inkrementierung bezw. Dekrementierung stattfinden soll; denn nach vollständiger Abarbeitung der inneren Schleife ist der Schleifenzähler ja

zerstört, so daß er beim nächsten Aufruf der inneren Schleife durch Abruf von R00 erst wieder regeneriert werden muß.

Für den Zähler der äußersten Schleife trifft das i.a. nicht zu, da dieser nur einmal durchlaufen wird.

6. Hier wird der aktuelle Y-Wert angezeigt, entsprechend 4.
7. In diesem Teil wird die eigentliche Funktionsberechnung durchgeführt, in diesem Fall einfach durch R-P.
8. Hier werden nacheinander die beiden Ergebnisse mit Text angezeigt.

```
      01♦LBL "SCHL"
1.    02 "XA ↑ XE ↑ DX"
      03 XEQ 00
      04 STO 01
2.    05 "YA ↑ YE ↑ DY"
      06 XEQ 00
      07 STO 00
3.    08♦LBL 01
4.    09 RCL 01
      10 INT
      11 STO 03
      12 "X="
      13 ARCL 03
      14 PROMPT
5.    15 RCL 00
      16 STO 02
      17♦LBL 02
6.    18 RCL 02
      19 INT
      20 "Y="
      21 ARCL X
      22 PROMPT
7.    23 RCL 03
      24 R-P
8.    25 "R="
      26 ARCL X
      27 PROMPT
      28 "∠="
      29 ARCL Y
      30 PROMPT
5.    31 ISG 02
      32 GTO 02
3.    33 ISG 01
      34 GTO 01
      35 RTN
Zu    36♦LBL 00
1.    37 PROMPT
und   38 1 E5
2.    39 /
      40 X<>Y
      41 1 E3
      42 /
      43 +
      44 +
      45 END
```

Die Beschränkung der Wertebereiche von X und Y auf nicht-negative Größen kann manchmal hinderlich sein. Unter der Voraussetzung, daß XE > XA und DX > 0 ist, kann der Bereich für X durch Addition einer geeigneten Konstanten zu XA und XE so verschoben werden, daß beide Werte ≥ 0 werden. Vor Anzeige und Übernahme des X-Wertes in die Funktionsberechnung (und entsprechend natürlich für Y) muß diese Verschiebung wieder rückgängig gemacht werden, denn sie diente ja nur dem fehlerfreien Arbeiten mit ISG. Eine entsprechende Transformation und Rücktransformation kann auch die Beschränkung bezüglich Ganzzahligkeit umgehen, aber das wird der mathematisch interessierte Leser sicher selbst herausfinden.

Klaus Werner Hoenow (11)

Olaf Gursch (■■0)
Grüntenstraße 29 b
1000 Berlin 42

Lieber Oliver und Mitglieder!

Ich habe das vorliegende Programmpaket, von dem ich überzeugt bin, daß es noch optimiert werden kann, in den letzten beiden Jahren im Urlaub benutzt. Es hat mir die 'Arbeit', d.h. die Verwaltung der Urlaubskasse wesentlich erleichtert. Es werden also mit meinem Programm all diejenigen angesprochen, die auch während des Urlaubs auf ihre Kasse achten wollen oder müssen. In der vorliegenden Form benötigt das Programmpaket zwei Memory Moduls (190 Register). Es läßt sich aber auch in einer abgemagerten Version mit nur einem Memory verwirklichen.
Da ich in beidem Jahren des Urlaubs mit dem Auto durch mehrere Länder unterwegs war, ist ein wichtiger Punkt für mich gewesen die Umrechnerei mit den Währungen meinem HP zu übertragen. Es entstand ein zweigeteiltes Programmpaket:
1) URLAUB mit den Unterprogrammen: CHANGE,UMR,RMU,WECHS,EINH.
2) TANKEN mit den Unterprogrammen: KM,DIST,KMKO.
Als Ergänzung dazu gibt es das Programm SUM und mit den HP-KEY NOTES Vol. 5 No. 1 flatterte das Programm TIMER auf meinen Tisch, so daß ich dieses auch noch mit aufnahm.
Im folgenden Abschnitt gebe ich eine Kurzbeschreibung der einzelnen Programme. Die Nummerierung findet sich im Programm-Listing im Anhang wieder.

1.1 URLAUB: Die Ausgaben werden mit Hilfe des Programmes in 5 Kategorien eingeteilt: Unterkunft, Ernährung, Andere Kosten, Fahrtkosten, Kleinkosten. Die Abspeicherung erfolgt in die Speicher 1 bis 5. Speicher Ø wird als Zwischenspeicher benutzt. Für die Benutzung mit anderen Währungen (Ausland) ist das Programm UMR als Unterprogramm erforderlich.

1.2 CHANGE }
1.3 WECHS }:Das Programm CHANGE berechnet den Wechselkurs und speichert die Einheit in Speicher 7 und den Kurs in Speicher 8. Als Unterprogramm wird WECHS aufgerufen. Dies führt aber nur zu einer vernünftigen Programmausführung, falls vorher in den Speichern 18 bis 21 maximal 4 Währungseinheiten gespeichert wurden. Bei Ausführung von CHANGE wird dann der aktuelle Wechselkurs für die entsprechende Währungseinheit in den Speichern 22 bis 25 festgehalten.

1.4 EINH: Mit EINH wird der Austausch der Währungseinheiten bzw. Kurse zwischen den von UMR benutzten Rechenspeichern 7,8 und den Speichern 18-21,22-25 vorgenommen, wenn man in ein neues Währungsgebiet einreist.

1.5 UMR: UMR rechnet ausländische Währungen in DM um. Hierzu muß die aktuelle Einheit der Währung in Speicher 7 stehen, der Kurs in Speicher 8. Befindet man sich längere Zeit in einem Währungsgebiet, kann man FLAG ØØ setzen. Dies verkürzt die Ausführung. Ist FLAG ØØ gelöscht, wird jedesmal nach einer Änderung der Einheit gefragt.

1.6 RMU: RMU leistet das Inverse zu UMR. Es rechnet DM in die in Speicher 7 vorhandene Einheit um.

1.7 SUM: Dieses Zusatzprogramm berechnet die Summe aller Ausgaben, die mit URLAUB gespeichert wurden. Am Anfang kann man einen Höchstbetrag der Ausgaben festsetzen, der dann als Vergleichswert dient und für jede Abfrage mit SUM den Restbetrag ausrechnet. Bei Überschreiten des Höchstbetrages erfolgt auch noch eine akustische Warnung mit BEEP.

2.1 TANKEN: Mit dem Programm TANKEN läßt sich die Kontrolle der Treibstoffkosten sowie des Verbrauchs durchführen.Es erfordert die Eingabe des Literpreises, sowie der Treibstoffmenge. In ausländischem Währungsgebiet ruft es ebenfalls UMR als Unterprogramm auf um die Berechnungen in DM durchzuführen. Die Treibstoffkosten werden automatisch zu den Fahrtkosten addiert (Speicher Ø4) und getrennt in Speicher 17 festgehalten.Die Programme KM und DIST werden aufgerufen um den durchschnittlichen Verbrauch mit der letzten Tankfüllung und während der gesamten Fahrt zu berechnen.

2.2 NACH: Dies ist ein Ergänzungsprogramm zu TANKEN. Sollte aus bestimmten Gründen, z.B. sehr teure Tankstelle, einmal der Tank nicht voll gefüllt werden, würde die Durchschnittsberechnung zu fehlerhaften Ergebnissen führen. Deshalb wird in diesem Fall mit Nach an Stelle von TANKEN gearbeitet.

2.3 KM: KM hält insbesondere den Kilometerstand zu Beginn der Reise fest und besorgt die Speicherverwaltung der einzelnen Kilometerstände bei der Durchschnittsberechnung von TANKEN.

2.4 DIST: Hiermit lassen sich jederzeit die bisher zurückgelegten Wegstrecken ermitteln.

2.5 KMKO: Dieses Programm berechnet die Kosten pro zurückgelegtem Kilometer während des Urlaubs. In Programmzeile Ø8 findet sich ein additiver Faktor. Dieser muß individuell bestimmt werden. Er berücksichtigt die jährlichen Fixkosten (STEUER,VERSICHERUNG, WARTUNG, RÜCKLAGE NEUWAGENKAUF,WERTMINDERUNG). Unter Umständen kann man hier die ADAC-TABELLE heranziehen (ADAC-MOTORWELT 12/80). Man muß dann aber die Treibstoffkosten herausrechnen. Für einen OPEL-ASCONA 1,6l ergibt sich dann monatl. Aufwand pro Kilometer ohne Treibstoffkosten bei 15000 km im Jahr: 0,2592DM/km.

Speicherbelegung beim Programmpaket:

Speicher:	Inhalt:
ØØ	Zwischenspeicher
Ø1	Unterkunft
Ø2	Ernährung
Ø3	Andere Kosten
Ø4	Fahrtkosten
Ø5	Kleinkosten
Ø6	Summe der Ausgaben nach Aufruf von Summe.

Speicher:	Inhalt:
Ø7	Währungseinheit
Ø8	Kurs
Ø9	Inverser Kurs
1Ø	Summe der getankten Liter Treibstoff
11	Zuletzt getankte Treibstoffmenge
12	Anfangs-Kilometerstand
13	Summe der gefahrenen Kilometer
14	Kilometer, die mit der letzten Tankfüllung erreicht wurden
15	Zwischenspeicher
16	Kilometerstand beim letzten Tanken
17	Summe der Treibstoffkosten
18	1. Währungseinheit
19	2. Währungseinheit
20	3. Währungseinheit
21	4. Währungseinheit
22	Kurs 1. Währungseinheit
23	Kurs 2. Währungseinheit
24	Kurs 3. Währungseinheit
25	Kurs 4. Währungseinheit
26	Höchstbetrag der Urlaubsausgaben

Für den Benutzer unbedingt erforderliche Kenhtnis der FLAGS:
Flag ØØ: Muß gesetzt werden, um bei längerem Aufenthalt in einem Währungsgebiet die Programme zu beschleunigen.
Flag Ø1: Wird automatisch gesetzt, wenn das Programm NACH benutzt wird.
Da ich keinen Drucker besitze ist das Listen mit der Schreibmaschine erfolgt. Alphanumerische Eingaben sind mit ' ' eingeschlossen.
Im Anhang folgen jetzt die Listings der Programme und dann ein Beispiel. Für das Beispiel gebe ich hier die Werte an:
In Deutschland getauscht: Für 363,23DM erhielt ich 2500 ÖS und für 103,75 DM erhielt ich 50000 Lire. Der Start erfolgte beim KM-Stand 82691. Getankt habe ich dann wie folgt:

Liter Preis	Liter	KM-STAND
1,32	31,83	83Ø65
1,489	1Ø,Ø7	Nachgetankt
1Ø,84 ÖS	28,04	836Ø8
98Ø Lire	1Ø,6	8372Ø

Ausgaben die das Programm URLAUB betreffen:
Ernährung DM 15.-, Unterkunft 175.-ÖS,Ernährung 1ØØ.-ÖS, Fahrtkosten (Mautgebühr) 17Ø ÖS , Kleinkosten 19,5Ø ÖS.

Viel Spaß beim Benutzen des Programms und während des nächsten Urlaubs!

Programm Listings:

1.1 URLAUB:

```
Ø1 LBL'URLAUB'
Ø2  CLST
Ø3 LBL a
Ø4  'PROGRAMM?'
Ø5  PROMPT
Ø6  ENTER
Ø7  STO ØØ
Ø8  X<= Ø?
Ø9  GTO a
1Ø  6
11  X<Y?
12  GTO a
13  CLX
14  FIX 2
15  1,ØØ5
16  STO 15
17 LBL b
18  RCL ØØ
19  RCL 15
2Ø  INT
21  X=Y?
22  GTO c
23  ISG 15
24  GTO b
25 LBL c
26  GTO IND X
27 LBL d
28  AVIEW
29  PSE
3Ø  DM?=1
31  PROMPT
32  ENTER
33  1
34  X=Y?
35  GTO e
36  SF Ø7
37  XEQ'UMR'
38 LBL e
39  CLA
4Ø  DM-BETRAG?
41  FC?C Ø7
42  PROMPT
43  RCL ØØ
44  X<>Y
45  ST+ IND Y
46  STOP
47  GTO a
48 LBL Ø1
49  'UNTERKUNFT'
5Ø  GTO d
51 LBL Ø2
52  'ERNAEHRUNG'
53  GTO d
54 LBL Ø3
55  'ANDERE KOSTEN'
56  GTO d
57 LBL Ø4
58  'FAHRTKOSTEN'
59  GTO d
6Ø LBL Ø5
61  KLEINKOSTEN
62  GTO d
63  END
```

1.2 CHANGE:

```
Ø1 LBL'CHANGE'
Ø2  'EINHEIT?'
Ø3  AON
Ø4  PROMPT
Ø5  ASTO Ø7
Ø6  AOFF
Ø7  'WIEVIEL DM?'
Ø8  PROMPT
Ø9  ENTER
1Ø  'WIEVIEL'
11  ├
12  ARCL Ø7
13  PROMPT
14  /
15  STO Ø8
16  CLA
17  '1'
18  ├
19  ARCL Ø7
2Ø  ├ =
21  ARCL Ø8
22  ├ DM
23  AVIEW
24  PSE
25  XEQ 'WECHS'
26  END
```

1.3 WECHS:

```
Ø1 LBL'WECHS'
Ø2  18
Ø3  STO ØØ
Ø4  RCL Ø7
Ø5 LBL a
Ø6  RCL IND ØØ
Ø7  X=Y?
Ø8  GTO b
Ø9  1
1Ø  ST+ ØØ
11  RDN
12  RDN
13  GTO a
14 LBL b
15  RCL ØØ
16  4
17  +
18  ST ØØ
19  FS? 1Ø
2Ø  RCL IND ØØ
21  FC? 1Ø
22  RCL Ø8
23  FS? 1Ø
24  STO Ø8
25  FC? 1Ø
26  STO IND ØØ
27  RTN
```

1.4 EINH:

```
Ø1 LBL'EINH'
Ø2  SF 1Ø
Ø3  'WELCHE WAEHRUNG?'
Ø4  AON
Ø5  PROMPT
Ø6  AOFF
Ø7  ASTO Ø7
Ø8  XEQ'WECHS'
Ø9  CF 1Ø
1Ø  RTN
11  END
```

1.5 UMR:

```
Ø1 LBL'UMR'
Ø2  FS? ØØ
Ø3  GTO a
Ø4  CLA
Ø5  'ZUR ZEIT:'
Ø6  |-
Ø7  ARCL Ø7
Ø8  AVIEW
Ø9  PSE
1Ø  'AENEDERUNG? ≠Ø'
11  PROMPT
12  X≠Ø?
13  XEQ'EINH'
14 LBL a
15  CLX
16  CLA
17  'WIEVIEL'
18  |-
19  ARCL Ø7
2Ø  PROMPT
21  STO 15
22 LBL'UM'
23  ENTER
24  RCL Ø8
25  *
26  FS? Ø7
27  RTN
28  CLA
29  ARCL Ø7
3Ø  |-:
31  ARCL 15
32  |-=
33  |-
34  ARCL X
35  AVIEW
36  STOP
37  END
```

1.6 RMU:

```
Ø1 LBL'RMU'
Ø2  RCL Ø8
Ø3  1/X
Ø4  STO Ø9
Ø5  CLA
Ø6  'WIEVIEL DM?'
Ø7  PROMPT
Ø8  STO ØØ
Ø9  RCL Ø9
1Ø  *
11  ARCL ØØ
12  |- DM=
13  |- ARCL X
14  |- ARCL Ø7
15  AVIEW
16  STOP
17  END
```

1.7 SUM:

```
Ø1 LBL'SUM'
Ø2  FS? Ø6
Ø3  GTO a
Ø4  'HOECHSTBETRAG?'
Ø5  PROMPT
Ø6  STO 26
Ø7  SF Ø6
Ø8 LBL a
Ø9  1,ØØ5
1Ø  STO 15
11  Ø
12  STO Ø6
13 LBL b
14  RCL IND 15
15  ST+ Ø6
16  ISG 15
17  GTO b
18  RCL Ø6
19  'BISHER AUSGEGEB'
2Ø  AVIEW
21  PSE
22  'EN:'
23  |-
24  ARCL X
25  |- DM
26  AVIEW
27  PSE
28  RCL 26
29  X<>Y
3Ø  -
31  'RESTBETRAG:'
32  |-
33  ARCL X
34  AVIEW
35  PSE
36  X<=Ø?
37  BEEP
38  END
```

2.1 TANKEN:

```
Ø1 LBL'TANKEN'
Ø2  'DM? =Ø'
Ø3  PROMPT
Ø4  X=Ø?
Ø5  SF Ø9
Ø6  '1L KOSTET?'
Ø7  PROMPT
Ø8  FS?C Ø9
Ø9  GTO c
1Ø  SF Ø7
11  XEQ'UM'
12 LBL c
13  ENTER
14  'WIEVIEL LITER?'
15  PROMPT
16  ST+ 11
17  ST+ 1Ø
18  *
19  PSE
20  ST+ Ø4
21  ST+ 17
22  PSE
23  FS? Ø1
24  GTO a
25  XEQ'KM'
26  RCL 14
27  ENTER
28  1ØØ
29  /
3Ø  RCL 11
31  X<>Y
32  /
33  CLA
34  'AUF 1ØØKM:'
35  |-
36  ARCL X
37  |- L
38  AVIEW
39  PSE
4Ø  SF Ø7
41  XEQ'DIST'
42  1ØØ
43  /
44  RCL 1Ø
45  X<>Y
46  /
47  CLA
48  'DURCHSCHNITTLIC'
49  AVIEW
5Ø  'HER VERBRAUCH:'
51  |-
52  ARCL X
53  |- L
54  AVIEW
55  PSE
56  Ø
57  STO 11
58 LBL a
59  RTN
6Ø  END
```

19 kann auch entfallen

2.2 NACH:

```
Ø1 LBL'NACH'
Ø2  SF Ø1
Ø3  'NICHT VOLL'
Ø4  AVIEW
Ø5  PSE
Ø6  'GETANKT? =Ø'
Ø7  X=Ø?
Ø8  PROMPT
Ø9  XEQ'TANKEN'
1Ø  CF Ø1
11  END
```

2.3 KM:

```
Ø1 LBL'KM'
Ø2  FS?Ø8
Ø3  GTO a
Ø4  'START BEI KM?'
Ø5  PROMPT
Ø6  STO 12
Ø7  STO 16
Ø8  SF Ø8
Ø9 LBL a
1Ø  'KM-STAND?'
11  PROMPT
12  STO 15
13  ENTER
14  RCL 16
15  -
16  STO 14
17  ST+ 13
18  RCL 15
19  STO 16
20  CLST
21  RTN
22  END
```

2.4 DIST:

```
Ø1 LBL'DIST'
Ø2  FS?C Ø7
Ø3  GTO a
Ø4  'KM-STAND?'
Ø5  PROMPT
Ø6  GTO b
Ø7 LBL a
Ø8  RCL 16
Ø9 LBL b
1Ø  ENTER
11  RCL 12
12  -
13  'GEFAHRENE KM:'
14  |-
15  ARCL X
16  AVIEW
17  PSE
18  RTN
19  END
```

2.5 KMKO:

```
Ø1 LBL'KMKO'
Ø2  XEQ'DIST'
Ø3  STO ØØ
Ø4  ENTER
Ø5  RCL 17
Ø6  X<>Y
Ø7  /
Ø8  Ø,2592†
Ø9  +
1Ø  FIX 4
11  '1 KM KOSTETE'
12  |-
13  ARCL X
14  |- DM
15  AVIEW
16  PSE
17  FIX 2
18  RCL ØØ
19  *
2Ø  'AUTOKOSTEN:'
21  |-
22  ARCL X
23  |- DM
24  AVIEW
25  STOP
26  END
```

† siehe Text

3. TIMER: s. auch HP KEY-NOTESVol.5 No.1

```
Ø1 LBL'TIMER'
Ø2 LBL A
Ø3  'SICHER FAHREN'
Ø4  AVIEW
Ø5 LBL Ø1
Ø6  STO ØØ
Ø7 LBL ØØ
Ø8  PSE
Ø9  1
1Ø  ST+ ØØ
11  GTO ØØ
12 LBL B
13  RCL ØØ
14  1ØØØØ
15  /
16  HR
17  1,5
18  *
19  1/X
2Ø  'AV='
21  ARCL X
22  |- KM/H
23  AVIEW
24  GTO Ø1
25  RTN
26  END
```

PROGRAMMBEISPIEL:

EINGABE	FUNKTION	ANZEIGE
	SIZE Ø27	
EINGABE DES PROGRAMMPAKETES		
	CLRG	
	ASN'TIMER'11	
	ASN'URLAUB'13	
	ASN'SUM'14	
	ASN'CHANGE'15	
	ASN'UMR'21	
	ASN'RMU'22	
	ASN'TANKEN'23	
	ASN'NACH'24	
	ASN'KM' 25	
	ASN'DIST'32	
	ASN'KMKO'33	
	ASN'EINH'34	
OES	ASTO 18	
LIRE	ASTO 19	
	XEQ'KM'(25)	START BEI KM?
82691	R/S	
	XEQ'CHANGE'(15)	EINHEIT?
OES	R/S	WIEVIEL DM?
362,23	R/S	WIEVIEL OES
25ØØ	R/S	1 OES=Ø,14 DM
	XEQ'CHANGE'(15)	EINHEIT?
LIRE	R/S	WIEVIEL DM?
1Ø3,75	R/S	WIEVIEL LIRE
5ØØØØ	R/S	1 LIRE=2,Ø8E-3DM
	XEQ'TANKEN'(23)	DM?=1
1	R/S	1 L KOSTET?
1,32	R/S	WIEVIEL LITER?
31,83	R/S	42,Ø2
		KM-STAND?
83Ø65	R/S	AUF 1ØØ KM:8,51L
		GEFAHRENE KM 374,ØØ
		DURCHSCHNITTLIC
		HER VERBRAUCH:8,51
	XEQ'URLAUB'(13)	PROGRAMM?
2	R/S	ERNAEHRUNG
		DM? =1
1	R/S	DM-BETRAG?
15	R/S	15,ØØ
	XEQ'NACH'(24)	NICHT VOLL
		GETANKT? =Ø
Ø	R/S	DM? =1
1	R/S	1 L KOSTET?
1,489	R/S	WIEVIEL LITER?
1Ø,Ø7	R/S	14,99
	XEQ'URLAUB'(13)	PROGRAMM?
1	R/S	UNTERKUNFT
		DM? =1
Ø	R/S	ZUR ZEIT: LIRE
		AENDERUNG? ≠ Ø
2	R/S	WELCHE WAEHRUNG
OES	R/S	WIEVIEL OES
175	R/S	25,36
	R/S	PROGRAMM?
2	R/S	ERNAEHRUNG
		DM? =1
Ø	R/S	ZUR ZEIT: OES
		AENDERUNG? ≠Ø
Ø	R/S	WIEVIEL OES

EINGABE	FUNKTION	ANZEIGE
1ØØ	R/S	14,49
	R/S	PROGRAMM?
4	R/S	FAHRTKOSTEN
		DM? =1
5	R/S	ZUR ZEIT:OES
		AENDERUNG? ≠Ø
Ø	R/S	WIEVIEL OES
17Ø	R/S	24,63
	SF ØØ	
	XEQ'TANKEN'(23)	DM? =1
2	R/S	1 L KOSTET?
1Ø,84	R/S	WIEVIEL LITER?
28,Ø4	R/S	44,Ø4
		KM-STAND?
836Ø8	R/S	AUF 1ØØ KM:7,Ø2 L
		GEFAHRENE KM 917,ØØ
		DURCHSCHNITTLIC
		HER VERBRAUCH: 7,63L
	XEQ'URLAUB'(13)	PROGRAMM?
5	R/S	KLEINKOSTEN
		DM? =1
4	R/S	WIEVIEL OES
19,5	R/S	2,83
	CF ØØ	
	XEQ'EINH'(34)	WELCHE WAEHRUNG?
LIRE	R/S	2,Ø8 -Ø3
	XEQ'TANKEN'(23)	DM? =1
Ø	R/S	1 L KOSTET?
98Ø	R/S	WIEVIEL LITER?
1Ø,6	R/S	21,56
		KM-STAND?
8372Ø	R/S	AUF 1ØØ KM:9,46 L
		GEFAHRENE KM 1Ø29,ØØ
		DURCHSCHNITTLIC
		HER VERBRAUCH: 7,83 L
	XEQ'UMR'(21)	ZUR ZEIT: LIRE
		AENDERUNG? ≠Ø
Ø	R/S	WIEVIEL LIRE
45ØØ	R/S	LIRE:45ØØ,ØØ=9,34
	XEQ'RMU'(22)	WIEVIEL DM?
112	R/S	112,ØØDM=53975,ØØ LIRE
	XEQ'SUM'(14)	HOECHSTBETRAG?
5ØØ	R/S	BISHER AUSGEGEB
		EN: 2Ø4,9Ø DM
		RESTBETRAG: 295,1Ø
	XEQ'KMKO'(33)	KM-STAND?
8372Ø	R/S	GEFAHRENE KM:1Ø29,ØØ
		1 KM KOSTETE:Ø,3783 DM
		AUTOKOSTEN: 389,32

BLEIBT NACHZUTRAGEN, WAS IHR EUCH LÄNGST GEDACHT HABT, IN KLAMMERN SIND DIE TASTEN-NUMMERN ANGEGEBEN LAUT ZUORDNUNG.

Widerstandsthermometer

DIN 43 760 / Okt. 80

Programm	Kommentar
01♦LBL "PT" 02 CF 01 03 SF 02 04 SF 21 05 CF 22	Haupt-Label, Einstellen der Flags
06 ,390802 07 STO 01	Koeffizient für lineares Glied
08 -58,0195 E-6 09 STO 02	Koeffizient für quadr. Glied
10 42,735 E -9 11 STO 03	Koeffizient für kubisches Glied
12 -100 13 / 14 STO 04	Koeffizient für biquadratisches Glied
15 FC? 55 16 GTO 01	Fehlt der Drucker?
17 ADV 18 SF 12 19 " Pt 100" 20 PRA 21 CF 12 22 " DIN 43 760 /" 23 "⊢ 10.80" 24 PRA	Zusammenstellen und Drucken des einleitenden Textes
25♦LBL 01 26 "TEMPERATUR ?" 27 1 28 PROMPT	Eingabe-Aufforderung für Temperatur Kennziffer setzen!
29 FC?C 22 30 GTO 00	Erfolgte keine Eingabe?
31 CF 00	benötigtes Flag in definierten Zustand bringen
32 850,01 33 X<=Y? 34 GTO 02 35 RDN 36 -200,01 37 X>Y? 38 GTO 02	Überschreitet die Eingabe den Geltungsbereich von DIN 43760 v. Okt. 80 für Pt 100 ?
39 RDN 40 ADV 41 XEQ 06	Stack ordnen Leerzeile Eingabe drucken
42♦LBL 03	Einstieg für Iteration
43 X<0? 44 SF 00	Liegt t-Wert unter 0°C?
45 ENTER↑ 46 ENTER↑ 47 ENTER↑	Stack füllen für Horner-schema
48 RCL 04 49 * 50 RCL 03 51 + 52 * 53 RCL 02 54 +	Ber. des biqu. und des kub. Gliedes. Quadr. Glied vorbereiten!
55 FC?C 00 56 X<> L	Wird nur quadr. und lin. benötigt?
57 * 58 RCL 01 59 + 60 *	Berechnung des quadr. und lin. Gliedes
61 100 62 +	Konstante addieren
63 FS? 01 64 GTO 04	War es eine Iteration?
65 GTO 07	Ausgabe-Routine
66♦LBL 02 67 BEEP 68 "WERT ILLEGAL" 69 AON 70 PSE 71 AOFF 72 GTO IND Z	Hinweis auf Eingabewerte, die außerhalb des Geltungsbereiches liegen. Rückkehr zu neuer Eingabe-Aufforderung
73♦LBL 00 74 "WIDERSTAND ?"	Vorbereitung zur Widerstands-Eingabe
75 0 76 STO 05 77 PROMPT	Iterationsregister löschen Kennziffer setzen
78 FC?C 22 79 GTO 01	Erfolgte keine Eingabe?
80 390,266 81 X<=Y? 82 GTO 02 83 RDN 84 18,491 85 X>Y? 86 GTO 02	Überschreitet die Eingabe den Geltungsbereich von DIN 43760 v. Okt. 80 für Pt 100 ?
87 RDN 88 ADV 89 XEQ 07	Stack ordnen Leerzeile Eingabe drucken

Pt 100 $-200°C \leq t \leq 850°C$

$R_t = f(t)$ und $t = g(R_t)$

Stroinski, September 81

366,367-81 prisma

Programm	Kommentar
90 STO 00	Eingabewert speichern
91◆LBL 05	Einstieg für Iteration
92 100 93 X>Y? 94 SF 01	Ist Iteration nötig? (R_t< 100 Ohm)
95 - 96 RCL 02 97 / 98 RCL 01 99 LASTX 100 / 101 2 102 / 103 X↑2 104 LASTX 105 RDN 106 + 107 SQRT 108 R↑ 109 + 110 CHS	Errechnen der Temperatur mit linearem und quadr. Glied. (für R_t< 100 Ohm nur Näherung, dann Iteration!)
111 FC? 01 112 GTO 06	Ist Iteration unnötig?
113 GTO 03	Absprung zur Iteration
114◆LBL 04 115 RCL 00 116 - 117 ABS	Prüfung des Iterationsergebnisses, Bildung der Differenz zur Eingabe
118 1 E-4 119 X>Y? 120 GTO 06	Ist Fehler genügend klein? Dann Ausgabe!
121 LASTX 122 ST+ 05 123 RCL 00 124 RCL 05 125 - 126 GTO 05	Vorbereitung der nächsten Iteration. Iterationsregister berichtigen! Neuer Start
127◆LBL 06 128 "T="	Ausgabe-Routine für Temperatur
129 FS?C 01 130 X<> T	Wurde Iteration durchgeführt? Wert nach X!
131 FIX 2 132 RND 133 ARCL X 134 "⊢ °C"	Formatieren, Runden, ins ALPHA-Reg. °C anhängen
135 0 136 GTO 08	Kennziffer setzen, Schluß-Routine!
137◆LBL 07 138 "R="	Ausgabe-Routine für Widerstand
139 FIX 3 140 RND 141 ARCL X 142 "⊢ OHM"	Formatieren, Runden, ins ALPHA-Reg. setzen, OHM anfügen
143 1	Kennziffer setzen
144◆LBL 08 145 X<>Y	Schlußroutine, Wert nach X!
146 FS? 55 147 PRA	Ist Drucker vorhanden? ALPHA-R. drucken!
148 FS?C 02 149 RTN	War es eine Eingabe?
150 SF 02 151 FC? 55 152 AVIEW 153 GTO IND Y 154 END	Falls Ergebnis, nächsten Durchlauf vorbereiten! Fehlt Drukker? Anzeigen! Zur zuletzt benutzten Eingabeaufforderung zurück!

347 BYTES
SIZE 006
TOT.REG. :056

Das Programm benutzt die Formeln und die Konstanten aus DIN 43 760 / Okt. 80

Die Auflösung (gegeben durch die Zeilen 118, 131 und 139) kann durch passende Änderung gröber oder feiner gemacht werden; die Zeilen 19 und 134 enthalten synthetische Befehle.

1.) Programm einlesen bzw. eingeben!
2.) XEQ PT
ohne Drucker: | mit Drucker: Pt 100 DIN 43 760 / 10. 80
Anzeige: "TEMPERATUR ?"
Mit R/S-Taste kann Aufforderung in "WIDERSTAND ?" geändert werden! Zyklischer Vorgang!
3.) Eingabe vornehmen, R/S-Taste drücken!
Anzeige des Ergebnisses | Druck der Eingabe, Druck des Ergebnisses
4.) R/S-Taste:
Anzeige wie vor voriger Eingabe!
5.) Weiter bei 3, bzw. R/S-Taste, wenn Eingabe gewechselt werden soll!

Widerstandsthermometer

DIN 43 760 / Okt.80

Programm	Kommentar
01◆LBL "NI"	Haupt-Label,
02 CF 01 03 SF 02 04 SF 21 05 CF 22	Einstellen der Flags
06 ,5485 07 STO 01	Koeffizient für lineares Glied
08 665 E-6 09 STO 02	Koeffizient für quadr. Glied
10 2,805 E-9 11 STO 03	Koeffizient für biquadratisches Glied
12 FC? 55 13 GTO 01	Fehlt der Drucker?
14 ADV 15 SF 12 16 " Ni 100" 17 PRA 18 CF 12 19 " DIN 43 760 /" 20 "⊦ 10.80" 21 PRA	Zusammenstellen und Drucken des einleitenden Textes
22◆LBL 01 23 "TEMPERATUR ?"	Eingabe-Aufforderung für Temperatur
24 1 25 PROMPT	Kennziffer setzen!
26 FC?C 22 27 GTO 00	Erfolgte keine Eingabe?
28 180,01 29 X<=Y? 30 GTO 02 31 RDN 32 -60,01 33 X>Y? 34 GTO 02	Überschreitet die Eingabe den Geltungsbereich von DIN 43760 v. Okt. 80 für Ni 100?
35 RDN 36 ADV 37 XEQ 06	Stack ordnen Leerzeile Eingabe drucken
38◆LBL 03	Einstieg für Iterationsverfahren

Programm	Kommentar
39 ENTER↑ 40 ENTER↑ 41 ENTER↑	Stack füllen für Horner-Schema
42 RCL 03 43 * 44 *	Berechnen des biquadr. Gliedes
45 RCL 02 46 + 47 *	Berechnen des quadr. Gliedes
48 RCL 01 49 + 50 *	Ber. des linearen Gl.
51 100 52 +	Konstante addieren
53 FS? 01 54 GTO 04	War es eine Iteration?
55 GTO 07	Ausgabe-Routine
56◆LBL 02 57 BEEP 58 "WERT ILLEGAL" 59 AON 60 PSE 61 AOFF 62 GTO IND Z	Hinweis auf Eingabewerte, die außerhalb des Geltungsbereiches liegen. Rückkehr zu neuer Eingabeaufforderung
63◆LBL 00 64 "WIDERSTAND ?"	Vorbereitung zur Widerstandseingabe-Aufforderung
65 0 66 STO 04 67 PROMPT	Iterationsregister löschen, Kennziffer setzen
68 FC?C 22 69 GTO 01	Erfolgte keine Eingabe?
70 223,23 71 X<=Y? 72 GTO 02 73 RDN 74 69,45 75 X>Y? 76 GTO 02	Überschreitet die Eingabe den Geltungsbereich von DIN 43760 v. Okt. 80 für Ni 100 ?
77 RDN 78 ADV 79 XEQ 07	Stack ordnen Leerzeile Eingabe drucken

Ni 100 -60°C ≦ t ≦ 180°C

$R_t = f(t)$ und $t = g(R_t)$

Stroinski, September 81

368,369-81 prisma

Programm	Bemerkung
80 STO 00	Eingabewert speichern
81◆LBL 05	Einstieg und
82 SF 01	Kennzeichnung der Iteration
83 100 84 - 85 RCL 02 86 / 87 RCL 01 88 LASTX 89 / 90 2 91 / 92 X↑2 93 LASTX 94 RDN 95 + 96 SQRT	Errechnen der Temperatur mit linearem und quadr. Glied als Näherung
97 R↑ 98 - 99 GTO 03 100◆LBL 04 101 RCL 00	Mit Näherung und exakter Formel neues R_t errechnen, Prüfung des Iterationsergebnisses.
102 - 103 ABS 104 1 E-4 105 X>Y? 106 GTO 06	Bildung der Differenz zur Eingabe. Ist Fehler genügend klein? Dann Ausgabe!
107 LASTX 108 ST+ 04 109 RCL 00 110 RCL 04 111 - 112 GTO 05	Vorbereitung der nächsten Iteration. Iterationsregister berichtigen! Neuer Start
113◆LBL 06 114 "T="	Ausgabe-Routine für Temperatur
115 FS?C 01 116 X<> T	Wurde Iteration durchgeführt? Wert nach XI
117 FIX 2 118 RND 119 ARCL X 120 "⊢ °C"	Formatieren Runden, ins ALPHA-Reg. °C anhängen
121 0 122 GTO 08	Kennziffer setzen zur Schluß-Routine!
123◆LBL 07 124 "R="	Ausgabe-Routine für Widerstand
125 FIX 3 126 RND 127 ARCL X 128 "⊢ OHM"	Formatieren Runden, ins ALPHA-Reg. setzen OHM anfügen
129 1	Kennziffer setzen
130◆LBL 08 131 X<>Y	Schluß-Routine Wert ins X-Reg.
132 FS? 55 133 PRA	Ist Drucker vorhanden? ALPHA-R drucken!
134 FS?C 02 135 RTN	War es eine Eingabe?
136 SF 02 137 FC? 55 138 AVIEW 139 GTO IND Y 140 END	Falls Ergebnis, nächsten Durchlauf vorbereiten. Fehlt Drucker? dann anzeigen! Zur zuletzt benutzten Eingabeaufforderung zurück!

313 BYTES
SIZE 005
TOT.REG. :050

Das Programm benutzt die Formel und die Konstanten aus DIN 43 760 / Okt. [illegible]

Die Auflösung (gegeben durch die Zeilen 104, 117 und 125) kann durch passende Änderung gröber oder feiner gemacht werden; die Zeilen 16 und 12o enthalten synthetische Befehle.

1.) Programm einlesen bzw. eingeben!

2.) XEQ NI

ohne Drucker:	mit Drucker:
	Ni 100 DIN 43 760 / [illegible]

Anzeige:
"TEMPERATUR ?"
Mit R/S-Taste kann Aufforderung in "WIDERSTAND ?" geändert werden! Zy[illegible]

3.) Eingabe vornehmen, R/S-Taste drücken!

Anzeige des Ergebnisses	Druck der Eingabe Druck des Ergebnisses

4.) R/S-Taste:
Anzeige wie vor voriger Eingabe!

5.) Weiter bei 3, bzw. R/S-Taste, wenn Eingabe gewechselt werden soll!

Der "Byte-Jumper" und der "Wolf".

Wohl jeder, der die "synthetische" Programmierung kennt und nutzt, kennt und nutzt auch den "Byte-Jumper". Manches Mitglied unseres Clubs kennt und nutzt auch den "Wolf"-Befehl, den Matthias Grabiak (76) in seinen Beiträgen über "synthetische" Programmierung im Weihnachtsheft 1980 vorgestellt hat. Möglicherweise wissen aber nicht alle, daß der "Byte-Jumper" und der "Wolf" nur zwei verschiedene Namen für die im Prinzip gleiche Funktion ist. Zwar ist nicht jeder "Byte-Jumper" ein "Wolf", aber jeder "Wolf" ist auch ein "Byte-Jumper".

Der "BJ" (BJ = Byte-Jumper) ist eine 2-Byte-Funktion, zusammengestellt aus einem "TEXT"-Befehl und einer beliebigen anderen 1-Byte-Funktion, mit Ausnahme des "Null"-Bytes. Es gibt 16 verschiedene "TEXT"-Befehle. Es sind die Befehle "TEXT Ø", "TEXT 1" usw. bis "TEXT 15". Daher gibt es auch mindestens 16 verschiedene Möglichkeiten, einen "TEXT"-Befehl für die Zusammenstellung zu einem "BJ" zu verwenden.

Die "TEXT"-Befehle sind Bytes mit den Kennzahlen (hex) FØ bis FF. Der "TEXT"-Befehl ist ein Byte, das dem Rechner signalisiert, daß die im Anschluß folgenden Bytes als Text zu deuten sind. Wenn wir, als Beispiel, in den Programm-Speicher die Zeile

```
...
03 ᵀAC
```

eingeben, stehen in der Zeile nicht 2 sondern 3 Bytes: F2,41,43. Das Byte F2 gibt an, daß 2 Text-Bytes folgen.

Der Entdecker des "BJ", William C. Wickes, gab der Funktion diesen Namen, weil sie den Rechner veranlaßt, von einer Byte-Deutung in die andere zu springen.

Nehmen wir das obige Beispiel. Wir haben die Zeilen:

```
...
02 X<>Y
03 ᵀAC
```

Wenn die Zeile 3 in der Anzeige steht, wir den PRGM-Modus verlassen und im RUN-Modus den "BJ" einsetzen, wird der Zusammenhang zwischen dem Byte F2 und den nachfolgenden Bytes zerrissen. Der Rechner deutet deshalb die Bytes 41 und 43 nicht mehr als "A" und "C", sondern als die Funktionen "-" und "/". Das können wir sehen, wenn wir wieder in den PRGM-Modus schalten. Wir sehen jetzt die Zeilen:

```
...
03 -
04 /
```

Wenn wir die Zeile 3 mit der "Backarrow"-Taste löschten, würde der Programm-Pointer zur Zeile 2 zurückspringen und wir könnten im Speicher sehen:

```
...
02 X<>Y
03 ᵀ‾C
```

Der Querstrich am oberen Rand der Anzeige in der Zeile 3 zwischen dem hochgestellten "T" un dem "C" ist der Display-Ausdruck für das "Null"-Byte.

Der "BJ", den Mr. Wickes entdeckte, bestand aus den Bytes F1 und 41. Deshalb wurde er später auch als "Standard-F1-BJ" bezeichnet, als man entdeckte, daß man auch alle anderen "TEXT"-Befehle zum Zusammenstellen eines "BJ" verwenden kann.

Vom "F3-BJ" an aufwärts, bezeichnet man den BJ in der anglo-amerikanischen Sprachwelt auch als "PRGM-Mode-BJ", als "Prefix-Maskor", oder als "Byte-Masker". Mit "masking" (maskieren) sollte der gleiche Vorgang umschrieben werden, den Matthias Grabiak meinte, als er dem "Wolf"-Befehl seinen Namen gab, weil diese Funktion Bytes "frißt".

Die von Hagen Klemp (73) im Prisma '81 auf den Seiten 235 und 236 angesprochenen Befehle "Wolf 1" bis "Wolf 9" könnte man auch als "F7-BJ" bis "FF-BJ" bezeichnen. Hagen nennt sie "Wolf 1" bis "Wolf 9" weil sie 1 bis 9 Bytes "fressen".

Was kann man mit dem "BJ" anfangen? Nun, wer Lust zum Probieren hat, wird sicher vielfältige Möglichkeiten für sich entdecken. Ich selber benutze die "BJ" für 3 verschiedene Anwendungen: 1.) Zum Zugriff auf die Status-Register. 2.) Um die überflüssige "1" vor den Exponenten-Zahlen zu entfernen. 3.) Um "synthetische" Text-Zeilen zu erzeugen.

Zur Zeit verwende ich 3 verschiedene "BJ" nebeneinander. Einen "F3-BJ" einen "F7-BJ" und einen "F8-BJ".

Nehmen wir einmal an, wir nehmen für jeden "BJ" als Postfix, also als 2. Byte, das Byte (hex) 5Ø. Das ist die Funktion "LN", bzw. der Buchstabe "P". Jeder "BJ" ist zwar eine 2-Byte-Funktion, aber wenn wir ihn im PRGM-Modus einsetzen, ohne manipulieren zu wollen, werden in die Programm-Zeile mehr als 2 Byte-Stellen eingefügt.

Beispiel 1: Wir setzen den "F3-BJ" ein:

```
...
02ᵀ-P-
```

Das Postfix-Byte steht immer an 2. Stelle, ein "Null"-Byte wurde zusätzlich eingefügt.

Beispiel 2: Wir setzen den "F7-BJ" ein:

```
...
02ᵀ-P-----
```

5 "Null"-Bytes wurden zusätzlich eingefügt.

Beispiel 3: Der "F8-BJ":

```
...
02ᵀ-P------
```

6 "Null"-Bytes wurden zusätzlich eingefügt.

Die Prefix-Bytes "F3", "F7" und "F8" bedeuten ja, daß im Anschluß noch 3, 7 bzw. 8 Text-Zeichen folgen sollen. Da tatsächlich aber immer nur 1 Zeichen folgt, werden die übrigen Stellen mit "Null"-Bytes aufgefüllt.

Wenn wir mit dem "F3-BJ" im PRGM-Modus ein "STO M" erzeugen wollen, müssen wir einen "Dummy" einsetzen. Als Dummy kann jeder 3-Byte-Befehl dienen. Beispielsweise "GTO 15" bis "GTO 99", oder "XEQ ØØ" bis "XEQ 99", oder "1 E" (einmal auf die EEX-Taste drücken). "1 E" umfaßt zwar eigentlich nur 2 Bytes, aber erst nach dem "Packen". Vorher besteht "1 E" aus den 3 Bytes ØØ,49,27. Ich bevorzuge den Dummy "1 E", weil er durch Druck auf eine einzige Taste erzeugt werden kann.

```
...
02 1 E            (ØØ,49,27)
03 STO IND 17     (91,91)
04 RDN            (75)
```

Der Dummy "1 E" hat die Aufgabe, Raum zu schaffen für den Einsatz

des "F3-BJ". Diese Aufgabe hat er durch die Eingabe bereits erfüllt. Jetzt wird er gelöscht und durch den "BJ" ersetzt. Sobald der "BJ" in der Zeile 2 steht, reißt er den Prefix 91 an sich und zerstört damit den Zusammenhang zwischen den beiden Bytes "STO" und "IND 17". Das 2. Byte wechselt in der Bedeutung von "IND 17" zu "STO", weil es als "IND 17" nicht isoliert im Programm-Speicher stehen kann. Da auch ein "STO" nicht allein stehen kann, reißt es das folgende Byte 75 an sich und in Zeile 3 steht jetzt "STO M".

```
...
02⊢P▮
03 STO M
```

Die "BJ"-Zeile 2 kann gelöscht werden.

Da der "F7-BJ" die gleiche Arbeit ohne den Einsatz eines Dummys machen kann, könnte man ihn als praktischer ansehen. Als Beispiel:

```
...
02 STO IND 17
03 RDN
```

Wir gehen zur Zeile 1 und setzen den "F7-BJ" ein. Jetzt sieht das Programm so aus:

```
...
02ᵀ-P----▮
03 STO M
```

Die Zeile 2 kann wieder gelöscht werden.

Nehmen wir an, wir wollen eine Zeile mit 3 "Null"-Bytes erzeugen, die im ALPHA-Register angehängt werden sollen (APPEND). Bei der Benutzung des "F3-BJ" müssen wir dafür folgendes kleine Programm eingeben:

```
...
02 XEQ Ø7  (EØ,Ø7)
03 1 E     (ØØ,49,27)
04ᵀ⊢AAA    (F4,7F,41,41,41)
```

Der gewählte Buchstabe "A" steht hier nur stellvertretend als Beispiel.

Zuerst löschen wir die Zeile 3 und setzen den "BJ" ein. Jetzt sieht das Programm so aus:

```
...
02 XEQ Ø7  (EØ,Ø7)
03ᵀ-P▮     (F3,ØØ,5Ø,F4)
04 CLD     (7F)
05 -       (41)
06 -       (41)
07 -       (41)
```

Das Byte F4 (TEXT 4) wurde gefressen, die Bedeutung des Bytes 7F wechselte von "⊢" zu "CLD" und aus den A's wurden die "-"'s. Die 3 "-"-Zeichen können wir entweder einzeln oder durch DEL ØØ3 gemeinsam löschen. Dafür müssen wir natürlich auf der Zeile 5 stehen. Jetzt gehen wir zur Zeile 2 und ersetzen wieder den dort stehenden Befehl durch den "F3-BJ". Nun haben wir folgendes Programm vor uns:

```
...
02ᵀ-P▮     (F3,ØØ,5Ø,F3)
03 LN      (5Ø)
04ᵀ⊢---    (F4,7F,ØØ,ØØ,ØØ)
```

Beim Ersetzen des "XEQ Ø7" durch den "F3-BJ" wurde das 1. Byte aus Zeile 3 "gefressen" und damit die anderen Bytes dieses Strings freigesetzt. Aus dem "P" wurde "LN" und das Byte F4 übernahm die nachfolgenden Bytes wieder als Text-Zeichen.

Bei der Verwendung des "F7-BJ" gibt es zwar auch Aufwand, aber der ist nicht ganz so groß. Hier heißt das Programm:

```
...
02 ⊢AAA    (F4,7F,41,41,41)
```

Ohne irgendetwas löschen zu müssen, gehen wir zur Zeile 1 und setzen sofort den "BJ" ein:

```
...
02 ⊤-P- - - -▩   (F7,ØØ,5Ø,ØØ,ØØ,ØØ,ØØ,F4)
03 CLD           (7F)
04 -             (41)
05 -             (41)
06 -             (41)
```

Auch hier ist wieder das Byte F4 "eingefangen" und die übrigen Bytes freigesetzt worden. Die 3 -'s löschen wir, gehen wieder zur Zeile 1 und setzen wiederum den F7-BJ ein.

```
...
02 ⊤-P- - - -▩   (F7,ØØ,5Ø,ØØ,ØØ,ØØ,ØØ,F7)
03 LN            (5Ø)
04 ⊤⊢- - -       (F4,7F,ØØ,ØØ,ØØ)
```

Bis jetzt war noch bei keiner Manipulation vom "F8-BJ" die Rede. Dieser kommt jetzt zu Ehren. Wir wollen einmal aus einem "1 E" ein "E", oder aus einem "1 E2" ein "E2", oder aus einem "1 E-3" ein "E-3" machen.

Zu Beginn erzählte ich, daß der Befehl "1 E" zwar an sich ein 2-Byte-Befehl ist, aber bei der Eingabe und vor einem "Packen" noch ein "Null"-Byte vorweg enthält. Diese Tatsache bedingt, daß beim Einsatz eines F3-BJ ein 2-Byte-Dummy statt eines 3-Byte-Dummys verwendet werden muß. 2-Byte-Befehle sind zum Beispiel "STO 16" bis "STO 99", oder "RCL 16" bis "RCL 99". Wollen wir ein "E" erzeugen, benötigen wir folgendes Eingangsprogramm:

```
...
02 STO 99
03 1 E
```

Den Befehl in Zeile 2 löschen wir und ersetzen ihn durch den F3-BJ

```
...
02 ⊤-P▩
03  E
```

Die Zeile 2 kann wieder gelöscht werden.

Beim Einsatz des F7-BJ brauchen wir zwar wieder keinen Dummy, können ihn aber auch nicht ohne weiteres einsetzen, weil er immer nur 1 Byte, das 1. Byte, frißt. Darum ist es in diesem Fall notwendig, zu "packen". Dafür gibt es 2 Möglichkeiten. Wir können GTO.. ausführen; das wäre aber nicht sehr ratsam, weil wir damit auch das Programm verlassen und erst einmal zurückkehren müssen. Die 2., hier anzuratende Möglichkeit ist, die Funktion PACK auszuführen. Das "Null"-Byte wird dabei mit weg-gepackt und wir können getrost den F7-BJ einsetzen. Die dabei entstehende Zeile wird anschließend natürlich wieder gelöscht.

Da das Packen manchmal recht lange dauert, bediene ich mich hier lieber des F8-BJ. Ihn kann man einsetzen, ohne packen zu müssen, weil er gleich 2 Bytes "frißt".

Es gibt noch eine Möglichkeit, durch Einsatz des F7-BJ und ohne packen ein x "E" zu schaffen. Dazu gibt man ein:

```
...
02 STO 27
```

Wenn wir jetzt auf Zeile 1 den F7-BJ einsetzen und er den Prefix STO "frißt", wechselt die Bedeutung des Postfix 27 ((hex) 1B) zum "E". Dieser Trick funktioniert aber nicht für "E2" oder "E-3" (als Beispiel).

Wer sich den NOP-Befehl FØ (TEXT Ø) erzeugen möchte, kann dies ganz bequem, in dem man

```
...
02 STO IND T    ((hex) 91,FØ)
```

eingibt. Wird auf Zeile 1 der F7-BJ eingesetzt, wird das Byte 91 "gefressen" und das Byte FØ freigesetzt. Der F3-BJ kann das natürlich auch, nur muß hier wieder vorher ein 3-Byte-Dummy eingesetzt werdon.

Ganz allgemein ist bei der Verwendung des F3-BJ darauf zu achten, daß vorher ein 3-Byte-Dummy eingesetzt werden muß, der bei der Verwendung eines F7-BJ nicht erforderlich ist.

Wir können nicht nur Befehle wie "RCL M" erzeugen, sondern auch solche wie "RCL IND M". Den Prefix "RCL" gewinnen wir aus dem Postfix "IND 16", und den Postfix "IND M" gewinnen wir aus dem Byte F5. Dieses Byte, das auch die Bedeutung "TEXT 5" hat, liefert uns der Rechner automatisch, wenn wir eine Gruppe von 5 Alpha-Zeichen in den Programm-Speicher eingeben; Beispielsweise "CCCCC". Wenn wir also ein "RCL IND M" erzeugen wollen, sieht unser Eingangsprogramm so aus:

```
...
02 STO IND 16
03 ⊤CCCCC
```

Wir gehen zur Zeile 1 und setzen den F7-BJ ein. Danach haben wir dies Programm vor uns:

```
...
02 ⊤-P----⊠
03 RCL IND M
04 /
05 /
06 /
07 /
08 /
```

Die BJ-Zeile 2 und die Funktionen "/" in den Zeilen 4 bis 8 können wir löschen.

Im Folgenden eine Liste der Postfix-Funktionen, aus denen wir Prefix-Funktionen gewinnen können.

Postfix wird	Prefix	Postfix wird	Prefix
IND 16	RCL	IND 17	STO
IND 18	ST+	IND 19	ST-
IND 20	STx	IND 21	ST/
IND 22	ISG	IND 23	DSE
IND 24	VIEW	IND 25	ΣREG

Postfix	wird Prefix	Postfix	wird Prefix
IND 26	ASTO	IND 27	ARCL
IND 28	FIX	IND 29	SCI
IND 30	ENG	IND 31	TONE
IND 40	SF	IND 41	CF
IND 42	FS?C	IND 43	FC?C
IND 44	FS?	IND 45	FC?
IND 78	X<>		

Hier eine Liste der 1-Byte-Funktionen, die wir für den direkten Zugriff auf die Status-Register verwenden können:

Funktion	wird Status-Register
RDN	M
LASTX	N
CLX	O
X=Y?	P
X≠Y?	Q
SIGN	⊢
X<=Ø?	a
MEAN	b
SDEV	c
AVIEW	d
CLD	e

Wenn wir indirekt über ein Status-Register arbeiten wollen, benötigen wir dafür ja einen "TEXT"-Befehl. Die "TEXT"-Befehle in der folgenden Liste bedeuten deshalb: TEXT 5 = Eingabe einer Gruppe von 5 Alpha-Zeichen; TEXT 6 = Eingabe einer Gruppe von 6 Alpha-Zeichen; usw.

aus	wird
TEXT 5	IND M
TEXT 6	IND N
TEXT 7	IND O
TEXT 8	IND P
TEXT 9	IND Q
TEXT 10	IND ⊢
TEXT 11	IND a
TEXT 12	IND b
TEXT 13	IND c
TEXT 14	IND d
TEXT 15	IND e

Für die Tasten-Zuordnung einer normalen Rechner-Funktion haben wir den Befehl ASN. Dieser Befehl nützt uns aber leider nichts, wenn wir einer Taste eine "synthetische" Funktion zuordnen wollen. Darum haben findige Köpfe das "KA"-Programm erfunden. "KA" bedeutet "Key Assignment", zu deutsch "Tasten-Zuordnung". Inzwischen gibt es mehrere unterschiedliche Versionen dieses Programms. Empfehlen möchte ich das "MK"-Programm. "MK" bedeutet "Multiple Key Assignment". Dieses Programm erlaubt uns nicht nur die normale Zuordnung, sondern auch die Zuordnung unter der Kontrolle eines eigenen Programms.

Um eine Taste mit einem BJ belegen zu können, müssen wir dem Programm mitteilen, welche Bytes verwendet werden sollen und welche Taste belegt werden soll. Jedes der 256 Bytes im Rechner hat eine "hexadezimale" und eine equivalente dezimale Kennzahl. Für die Eingabe in das "KA"- bzw. "MK"-Programm benötigen wir nun die dezimale Kennzahl. Die Aufstellung aller verwendbaren Bytes und die ~~dazugehörigen Kennzahlen finden~~

dazugehörigen Kennzahlen finden wir in der HP41C-Byte-Tabelle. Jeder, der sich mit "synthetischer" Programmierung befaßt, sollte sich deshalb eine solche Tabelle besorgen.

Für die Belegung einer Taste müssen 3 Zahlen in das Programm eingegeben werden: 1. Der Prefix, 2. der Postfix, 3. der Tastenkode. In meinen Beispielen habe ich einen F3-, einen F7- und einen F8-BJ verwendet. Die Prefix-Zahlen dafür sind: F3 = 243, F7 = 247, F8 = 248. Bei allen 3 BJ-Beispielen habe ich einheitlich das Postfix-Byte 5Ø verwendet. Das Byte 5Ø ist dezimal 8Ø.

Die "KA"-Programme und das "MK"-Programm können beim Karten-Copy-Service bezogen werden. Mindestens für das "MK"-Programm existiert eine mehrseitige Bedienungsanleitung. Wer meiner Empfehlung folgen und das "MK"-Programm haben möchte, schicke bitte mindestens 4 leere Magnetkarten an den Service. Es empfiehlt sich, einen selbst adressierten, frankierten Umschlag mitzuschicken. Der Umschlag sollte Größe DIN C 4, mindestens aber DIN C 5 haben. Das Freiporto sollte DM 1,40 betragen. Der Service ist über 2 Adressen zu erreichen, die im Prisma '81 auf den Seiten 239 und 240 zu finden sind.

Niels Nöhren (81)
Kielort 16
D-2000 Norderstedt
09.10.1981


```
NACHSTEHENDES PRGR. PLOT
TET DAS SPIELFELD FUER
DAS U-BOOT JAGD PRGR VON
OLIVER (S. PRISMA 1980)
MILOSLAV FOLPRECHT

              PRP "SF"

 01+LBL "SF"
FIX 0  CF 29  6  SKPCHR
"U-BOOT JAGD"  ACA
PRBUF  ADV  XEQ 01  ,009

 12+LBL 00
CLA  INT  ARCL X  XEQ 03
ARCL X  LASTX  "PRA
ISG X  GTO 00

 22+LBL 01
1  SKPCHR  ,009

 26+LBL 02
CLA  " "  INT  ARCL X
ACA  LASTX  ISG X
GTO 02  PRBUF  RTN

 37+LBL 03
"-I I I I I I I "
"-I I I I"  END
```

```
      U-BOOT JAGD

  0 1 2 3 4 5 6 7 8 9
0I I I I I I I I I I I0
1I I I I I I I I·I I I1
2I I I I I I I I I I I2
3I I I I I I I I I I I3
4I I I I I I I I I I I4
5I I I I I I I I I I I5
6I I I I I I I I I I I6
7I I I I I I I I I I I7
8I I I I I I I I I I I8
9I I I I I I I I I I I9
  0 1 2 3 4 5 6 7 8 9
```

Hilfsmittel [illegible]r synthetischen Programmierung.

Mancher, der sich mit synthetischer Programmierung beschäftigt, kennt das LB(Load Bytes)-Programm. Viele kennen das KA(Key-Assignment)-Programm. Viele wissen auch um die Existenz des PPC-ROM's.

Der U.S.amerikanische Anwender-Club PPC hat dafür gesorgt, daß in das ROM von beiden Programmen je eine qualitativ besonders hochwertige Version hineinkommt. Um seinen Mitgliedern schon vor Auslieferung des ROM's die Möglichkeit zu bieten, diese Programme benutzen zu können, wurden sie in Form von Bar-Kode zusammen mit einer guten Benutzer-Anweisung im PPC CJ V8N2P34-40 abgedruckt.

Die Programme und die Anleitung wurden vom PPC-Mitglied Clifford Stern geschrieben. Bevor die Programme in der vorliegenden Form gedruckt wurden, wurden sie vom PPC-Mitglied Keith Jarett in einigen Punkten modifiziert. Clifford Stern ist ein Könner auf dem Gebiet der synthetischen Programmierung. Keith Jarett jedoch hat ein so tiefes Verständnis für die inneren Zusammenhänge der synthetischen Programmierung entwickelt, daß man ihn geradezu als Genie einstufen könnte. Soweit die Quell-Informationen.

Da nicht alle Clubfreunde die amerikanische Sprache verstehen, oder zumindest glauben, nicht genügend Kenntnisse zu besitzen, habe ich mich bemüht, die Anleitung zu übersetzen, so gut es mir eben möglich war. Dadurch können alle Interessierten die Programme nutzen. Außerdem habe ich noch eine Byte-Tabelle hinzugefügt, weil die zur synthetischen Programmierung gehört, wie das Amen in die Kirche. Wer das beigefügte Listing nicht abtippen möchte (das dürfte auch ein wenig schwierig und langwierig werden), kann sich von unserem Karten-Copy-Service beschriebene Magnetkarten zuschicken lassen.

Wichtig! Verlangt nicht einfach die beschriebenen Karten, sondern schickt Blanko-Karten hin. Außerdem sollte es sich von selbst verstehen, daß ein "SAFU" (selbst adressierter frankierter Umschlag) mitgeschickt wird. Weder die Magnetkarten, noch die Briefumschläge, noch das Porto können vom Club spendiert werden.

Für das LB-Programm sind 4 Karten erforderlich. Für das MK(KA)-Programm ebenfalls 4. Wer beide haben möchte, kommt mit 7 Karten aus. Das liegt daran, weil die Programme in 3 Gruppen aufgeteilt wurden. Gruppe 1 besteht aus Unterprogramm-Routinen, die von beiden Hauptgruppen, LB und MK, gemeinsam benutzt werden. Gruppe 1 (DC) paßt auf 1 Karte. Gruppe 2 (LB) und 3 (MK) passen auf je 3 Karten.

Interessierte schreiben bitte an:

Walter Pieperhoff oder Ralf Pfeifer
Walb. Osth. Wallstr. 10 — Rubensstr. 5
4770 Soest — 5000 Köln 50
Tel. (02921) 28 95 — Tel. (0221) 35 20 34

Niels Nöhren (81)
02.09.1981

Bei Magnetkartenanforderungen unbedingt Seite 405 vorher lesen !

Zu den am meisten benötigten Hilfsmitteln in der synthetischen Programmierung gehören das LB- (Load Bytes) und das MK- (Make Key Assignments) Programm. Mit dem Programm LB können wir jede gewünschte Sequenz Bytes direkt in den Programm-Speicher laden, ohne Byte-Jumping, oder andere Tricks anwenden zu müssen. Dieses Programm beinhaltet 2 non-prompting (aufforderungslose) Versionen (L- und -B), die es uns gestatten, unter der Kontrolle eines eigenen Programmes, Bytes zu laden. Mit dem MK können wir jede gewünschte Funktion einer beliebigen Taste zuordnen. In diesem Programm sind ebenfalls 2 Versionen (1K und +K) enthalten, die die Zuordnung unter Programm-Kontrolle gestatten.

Beide Programme enthalten Sicherheits-Vorkehrungen, damit nicht belegte Teile des Speichers überschrieben werden und informieren den Benutzer, wieviele Register verfügbar sind. LB erlaubt Fehlerkorrekturen; MK schützt vorhandene Zuordnungen, überprüft auf nichtexistierende oder benutzte Tasten usw.

Die Programme sind in 3 Gruppen aufgeteilt; jede mit einem eigenen END. Die 1. Gruppe (DC-Gruppe) enthält Unterprogramm-Routinen (DC, 2D, OM und VA), die von beiden, von LB und MK verwendet werden. Die 2. Gruppe (LB-Gruppe) enthält die Routinen LB, L-, und -B zusammen mit den Unterroutinen XD und QR, die von LB, nicht aber von MK benötigt werden. Die 3. Gruppe (MK-Gruppe) enthält die Routinen MK, 1K, +K und die Unterroutinen LF und E?. Es können alle 3 Gruppen geladen werden. Sollen LB oder MK allein verwendet werden, muß DC auf jeden Fall mit geladen werden. Wichtig! Nach dem Laden jeder Gruppe muß GTO.. ausgeführt werden, damit am Ende jeder Gruppe ein END gebildet wird und vor LB oder MK muß auch ein END stehen. Sollen die Gruppen in einen leeren Speicher geladen werden, ist anzuraten, DC als 1. Gruppe zu laden.

Benutzeranweisungen für LB:

1. Vor LB muß ein END im Programm-Speicher stehen. Der SIZE muß 012 oder größer sein.

2. Wir gehen zu dem Punkt im Programm-Speicher, an dem die Bytes eingefügt werden sollen (muß nicht am Ende des Programm-Speichers sein) und geben das Folgende im PRGM-Modus ein:

   LBL "++" + + + ... + + + XEQ "LB"

   (Die +'s bereiten den Raum im Speicher für das Einfügen der Bytes vor und werden außerdem zur Raumüberprüfung gebraucht, deshalb verwenden wir keine ENTER oder etwas anderes an Stelle der + Instruktionen.) Damit genügend Raum für n Bytes vorhanden ist, sollte die Anzahl der "+'s" "n'+6" betragen, wobei "n'" das kleinste Vielfache von 7 größer oder gleich "n" ist. (Z.B., 13 "+'s" sind erforderlich zum Einfügen von 1-7 Bytes, 20 "+'s" für 8-14 Bytes, usw.) Man kann auch einfach für jedes Byte ein "+" plus 12 extra "+'s" programmieren. Überflüssige "+'s" werden automatisch durch Null-Bytes ersetzt.

2.1. Der Speicher braucht nicht gepacked zu werden.

3. Steht das Programm noch auf XEQ "LB", so verlassen wir den PRGM-Modus und drücken R/S. (Es kann von jedem Punkt aus zwischen LBL "++" und XEQ "LB" gestartet werden. Man kann LB auch vom Tastenfeld aus starten.) Nach ein paar Sekunden stoppt das Programm mit der Aufforderung "≠≠1 OF m?", wobei "m" die maximale Anzahl der Bytes darstellt, die geladen werden können, aufgrund der Anzahl "+'s", die programmiert wurden.

3.1. Ist die angezeigte Zahl "m" nicht groß genug, führen wir GTO "++" aus und fügen mehr "+'s" ein. Haben wir von vornherein nicht genug "+'s" programmiert, wird der Rechner "SST, MORE +'s" auffordern. Nach Ausführung von SST sind wir bei LBL "++" und können mehr "+'s" einfügen. Danach kann neu gestartet werden.

4. Wir geben das dezimale Equivalent des einzufügenden Bytes im Nicht-ALPHA-Modus ein, oder die Hex-Zahl des Bytes im ALPHA-Modus. Nach jeder Eingabe drücken wir R/S. Um das dezimale oder Hex Equivalent eines zu ladenden Bytes zu ermitteln, ziehen wir die Byte-Tabelle zu Rate. Dezimel oder Hex Eingaben können gemischt werden. Jede Eingabe wird nach ihrer Art richtig verarbeitet. Für die Hex Eingabe geben wir aber nur 2 und wirklich nur 2 Ziffern ein.

5. Um das Byte-Loading zu beenden, drücken wir R/S ohne vorheriger Eingabe. Wenn so viele Bytes geladen wurden, wie geladen werden konnten, endet der Prozess automatisch. Jetzt erscheint die Aufforderung "SST, DEL 00p", "p" wird eine Zahl zwischen 1 und 7 sein. Wir führen SST aus, gehen in den PRGM-Modus (wir sehen LBL "++") und führen DEL 00p aus. Dies vernichtet LBL "++" und die restlichen "+'s".

6. Hinter der kreierten Zeile stehen möglicherweise noch "+'s", gefolgt von XEQ "LB". Sollen weitere Zeilen geschaffen werden, können sie stehen bleiben, ansonsten gelöscht werden. Im X-Register steht eine Zahl "p,00q". "p" war die zu löschende Zeilenzahl vor der kreierten Zeile, "q" ist die zu löschende Zeilenzahl hinter der kreierten Zeile.

7. Wenn wir einen Fehler in der Eingabe bemerken und bereits R/S gedrückt haben, brauchen wir nur XEQ 03 auszuführen. Oder, wenn der Rechner im ALPHA-Modus steht, können wir irgendeinen Buchstaben eingeben und R/S drücken. Die jeweils letzte Eingabe wird damit rückgängig gemacht. So kann man alle Eingaben, eine nach der anderen, rückgängig machen. Dies gilt, solange LB noch nicht verlassen wurde. Eine negative Eingabe hat den gleichen Effekt. Dies ist wichtig für Benutzung von LB unter Programm-Kontrolle.

8. Sollten wir vergessen haben, welches die letzte Eingabe war, so werden wir nach XEQ 01 daran erinnert. Ist das Byte-Loading bereits beendet, haben aber LB durch SST noch nicht verlassen, so hat XEQ 01 die gleiche Wirkung wie XEQ 03.

Warnungen: (a) Es liegt in der Verantwortung des Benutzers, dafür zu sorgen, daß genug Raum vorhanden ist, um eine Multi-Byte-Zeile beenden zu können. Wenn wir beginnen, eine Text-Zeile mit einer Länge von 15 Zeichen zu kreieren und der Vorgang wird beendet, ohne daß die letzten beiden Zeichen untergebracht werden können, dann müssen wir damit rechnen, daß einiges aus einem anderen Programm, sogar ENDs verschluckt werden. (b) Während des Byte-Loadings darf weder gepacked noch der SIZE verändert werden. (c) LB darf nicht verlassen werden, ohne die Beendigungs-Prozedur (Punkt 5) durchgeführt zu haben, insbesondere, wenn Fehler korrigiert wurden. (d) Während des Byte-Loading dürfen wir sowohl den Stack als auch die Register 00-05 benutzen, aber nicht die Register 06-11, weil die von LB intern verwendet werden.

Beispiel zum Gebrauch: Nehmen wir an, wir wollen eine Text-Zeile kreieren, die so aussieht: ᵀ⊢‾ҲµC (das sind 4 Zeichen, die wir anhängen (Append) wollen, mit den Hex-Kodes 00, 01, 0C und 40), und diese Zeile soll eingefügt werden nach dem TONE 9 im folgenden

Programm:

LBL "TEST" TONE 9 AVIEW END

Wir gehen davon aus, daß dieses Programm das 1. Programm-Speicher ist.

1. Wenn wir auf der Byte-Tabelle nachsehen, stellen wir fest, daß die Bytes folgende Kennzahlen haben:

   F5 7F 00 01 0C 40 (hex), oder
   245 127 0 1 12 64 (dezimal).

2. Wir gehen zu TONE 9 und, im PRGM-Modus, tippen ein:

   LBL "++" (13 +'s) XEQ "LB"

3. Wir verlassen den PRGM-Modus und drücken R/S. Nach ein paar Sekunden sehen wir die Aufforderung "≠≠1 OF 7?".

4. Wir geben ein: 245, R/S, 127, R/S, 0, R/S, 1, R/S, 12, R/S, 64, R/S, (jetzt müßte "≠≠7 OF 7?" zu sehen sein). Als Alternative könnten wir im ALPHA-Modus eingeben: F5, R/S, 7F, R/S, 00, R/S, 01, R/S, 0C, R/S, 40, R/S. Oder wir können die dezimalen und Hex Eingaben mischen (nur darf der Modus nicht zwischen der Eingabe und R/S gewechselt werden). Die Hex Eingaben werden langsamer verarbeitet, weil sie in das dezimale Equivalent umgerechnet werden müssen.

5. Wir drücken R/S in beiden Modi, ohne vorher eine Eingabe gemacht zu haben, um den Prozess zu beenden. Wir sehen die Aufforderung "SST, DEL 006". Wir drücken SST (die Zahl 6,002 in X verdient Beachtung), gehen in den PRGM-Modus und führen DEL 006 aus. Nach Druck auf SST können wir die kreierte Text-Zeile sehen. Der Bruch-Anteil ,002 in X bedeutet, daß nach der Text-Zeile noch 2 Befehle zu löschen wären (+, XEQ "LB").

Anweisungen für L- und -B: Diese Routinen gestatten das Byte-Loading unter Kontrolle eigener Programme. Beispiele sollen hier nicht gegeben werden, aber generelle Regeln!

1. Wir programmieren XEQ "L-" in dem Programm, das das Byte-Loading kontrollieren soll. Dieser Befehl initialisiert den Prozess und kehrt zum Kontroll-Programm zurück.

2. -B produziert die Bytes und orientiert sich an einer Zahl, die in X stehen muß. Es müssen dezimale und dürfen keine Hex Zahlen sein. Es wird immer nur 1 Byte zur Zeit produziert.

3. Um den Prozess zu beenden, muß die Befehlsfplge CF 09, XEQ "-B" im Programm stehen. Danach erscheint "SST, DEL 00p".

4. Bevor das Kontroll-Programm gestartet wird, überprüfen wir auf SIZE 012 und schaffen dort Raum im Speicher, wo die Bytes geladen werden sollen, in dem wir wie üblich programmieren: LBL "++", eine Reihe von "+'s", XEQ "LB".

5. Wir verlassen den PRGM-Modus und starten das eigene Programm, anstatt durch R/S den Byte-Loader zu starten.

6. Die Ausführung wird wie gehabt beendet mit der Aufforderung: "SST, DEL 00p", so daß das Weitere wie beim normalen Byte-Loading abläuft.

7. Soll unser Programm ein falsches oder unerwünschtes Byte korrigieren, so geschieht es, in dem eine negative Zahl nach X gegeben und "-B" ausgeführt wird.

Warnung: "-B" darf nicht aufgerufen werden, ohne daß vorher erst

"L-" aufgerufen wurde. Einige Flag und andere Sicherheitsvorkehrungen werden durch "L-" aktiviert, weil "-B" allein für sich ausgeführt zu MEMORY LOST führen kann, oder zumindest andere Programme beschädigt oder vernichtet.

Anweisungen für MK:

1. Wie bei LB muß vor MK ein END im Programm-Speicher stehen und der Size muß 012 oder größer sein.

2. Wir starten durch XEQ "MK". Das Programm überprüft das Key-Assignment-Register von oben bis unten und zeigt in einer PSE an, wieviele Register noch frei sind. Wenn wir diese Zahl verdoppeln, wissen wir die mögliche Anzahl von Zuordnungen.

2.1. Wenn "NO ROOM" angezeigt wird, müßten wir entweder einige Zuordnungen löschen, oder den SIZE reduzieren. Anschließend kann durch R/S neu gestartet werden. Es gäbe auch noch die Möglichkeit, andere Programme zu löschen, oder das ROM-Programm PK aufzurufen, um das Key-Assignment-Register zu packen.

3. Nachdem die Anzahl der freien Register genannt wurde, fordert MK "PRE/POST/KEY" auf. Wir geben das dezimale Equivalent für das 1. Byte (Prefix) ein, drücken Enter, das 2. Byte (Postfix), Enter, und den Tasten-Kode. Dann starten wir die Zuordnung mit R/S. (Z.B. 159, Enter, 26, Enter, -81, R/S; um TONE 26 auf die geshiftete Taste in der Zeile 8, Spalte 1 zu legen.) Wenn die Belegung richtig vollzogen wurde, fordert MK zur nächsten Eingabe auf. (Die Zuordnungen werden nicht gezählt, müssen also nicht paarweise eingegeben werden, wie bei anderen KA-Programmen.) Der Stack wird vor Eingabe-Aufforderung gelöscht, damit der Prefix Null ist, falls versehentlich nur Postfix und Tasten-Kode eingegeben werden.

4. Wenn eine Null für den Tasten-Kode eingegeben wird, oder R/S ohne Eingabe gedrückt wird, zeigt der Rechner die Anzahl der noch freien Register an, um dann wieder zur Eingabe aufzufordern.

5. Wenn wir keine Zuordnungen mehr machen wollen, können wir entweder den Rechner ausschalten, oder an irgendeinen anderen Punkt des Programm-Speichers gehen. Bei MK ist keine Beendigungs-Prozedur notwendig.

6. Die Anzeige "KEY TAKEN", gefolgt von dem PROMPT "KEYCODE?" bedeutet, daß die gewählte Taste bereits belegt ist. Entweder löschen wir die vorhandene Belegung, oder wir wählen einen anderen Tasten-Kode und drücken R/S um die neue Belegung zu vollziehen. "NO SUCH KEY" gefolgt von "KEYCODE?" bedeutet, daß versucht wurde, eine nicht vorhandene Taste zu belegen.

   Nach jeder Fehler-Anzeige steht der ursprüngliche, falsche Tasten-Kode in X, so daß er wieder verwendet wird, wenn R/S ohne Neueingabe gedrückt wird. Bei einer neuen, richtigen Eingabe wird die Belegung vollzogen. Bei einer Null als Eingabe, werden wieder die noch freien Register angezeigt und anschließend zur neuen Gesamt-Eingabe aufgefordert.

7. Die Anzeige "DONE, NO MORE" bedeutet, daß die letzte Belegung getätigt wurde und jetzt kein Raum mehr für neue Belegungen vorhanden ist. Falls trotzdem neue Belegungen gemacht werden sollen, müssen wir wie bei Punkt 2.1. handeln.

8. Nach jedem Stopp wegen einer Fehler-Anzeige (Punkt 2.1., 6. oder 7.), oder, falls der Rechner aus- und wieder eingeschaltet wurde, überprüft er das Key-Assignment-Register, um sicherzustellen, daß eine neue Belegung ohne Überlappung oder Lücke geschieht.

(Durch den Test von Flag 20 wird entschieden, ob die Register überprüft werden müssen. Ist das Flag gelöscht, werden die Register überprüft.)

Warnung: (a) Falls zwischen 2 Tasten-Belegungen gepackt, oder der SIZE geändert werden muß, schalten wir den Rechner aus und wieder ein (oder löschen Flag 20), um dem Programm zu signalisieren, daß die Register überprüft werden müssen. (b) Es darf nichts in die Daten-Register 09, 10 oder 11 abgespeichert werden, oder der Zustand der Flags 07, 09, 10 oder 20 zwischen 2 Belegungen gewechselt werden. Wie bei LB werden von MK die Register 06-11 intern benutzt. R 06, R 07 und R 08 enthalten den Prefix, Postfix und Tasten-Kode für die zuletzt eingegebene Belegung; R 09 enthält den Index für die indirekte Speicherung der nächsten Belegung und R 10 enthält den Inhalt des c-Registers. Wenn Flag 10 vom Benutzer gesetzt wurde, enthält R 11 die 1. Belegung eines Tasten-Belegungs-Paares.

Anweisungen für 1K und +K: Das Folgende ermöglicht es, Tasten-Belegungen unter Programm-Kontrolle zu machen.

1. Der SIZE muß 012 oder größer sein.

2. Wenn wir unter Programm-Kontrolle eine einzige Belegung machen wollen, müssen wir dafür sorgen, daß der Prefix in Z, der Postfix in Y und der Tasten-Kode in X steht, bevor "1K" als Unterprogramm (XEQ "1K") aufgerufen wird. Wenn die Belegungseingabe einen Fehler enthält, wird das Programm mit einer Fehler-Anzeige anhalten, vorausgesetzt, wir haben nicht Flag 25 gesetzt, bevor "1K" aufgerufen wurde. Wurde Flag 25 gesetzt, wird bei einem Eingabe-Fehler die Belegung nicht durchgeführt. Im ALPHA-Register kann nachgelesen werden, um welchen Fehler es sich gehandelt hat.

3. Sollen mehrere Belegungen durchgeführt werden, wird für die 1. "1K" und für jede weitere "+K" aufgerufen. Vor jedem Aufruf müssen die entsprechenden Daten in Z, Y und X stehen. "+K" tut das gleiche wie "1K", aber es überprüft nicht die Register (wenn Flag 20 gesetzt ist); dadurch erfolgen die Belegungen wesentlich schneller. Ist jedoch Flag 20 gelöscht, überprüft "+K" genau wie "1K" die Register. (Nach einer Register-Überprüfung ist Flag 20 jedoch stets gesetzt.) Ist das Kontroll-Programm so umfangreich, daß es unbequem wird, "1K" für die 1. und "+K" für jede weitere Belegung aufzurufen, dann kann "+K" für alle Belegungen verwendet werden. Es müssen nur jedesmal Flag 07 gesetzt und Flag 20 gelöscht werden, bevor "+K" aufgerufen wird. (Flag 07 verhindert das PROMPT.)

4. "1K" und "+K" können auch vom Tastenfeld aus gestartet werden, nachdem die entsprechenden Daten in den Stack gebracht wurden. Wird "+K" ausgeführt, wird es die PROMPTing-Version verwenden, falls vorher "MK" ausgeführt wurde (Flag 20 ist gelöscht), oder die non-PROMPTing-Version, falls "1K" vorher ausgeführt wurde (Flag 20 ist gesetzt).

Warnung: Wir dürfen "+K" nicht allein benutzen (ohne daß vorher "MK" oder "1K" ausgeführt wurde) ohne sicher zu sein, daß Flag 20 gelöscht ist und wir müssen den Zustand von Flag 07 überprüfen, um sicher zu sein, daß die richtige Version abläuft. Die gleiche Warnugg (Flag 20 muß gelöscht sein) gilt, wenn der SIZE geändert wurde, Tasten-Belegungen manuell ausgeführt oder gelöscht wurden, das Key-Assignment-Register gepacked wurde oder die Inhalte der Register 09-11 verändert wurden.

Verwendete Unterprogramme: Das Folgende ist eine kurze Beschreibung der Routinen, die von "LB" und "MK" verwendet werden.

DC (Decimal to Character): Wandelt das in X stehende, dezimale Equivalent eines Bytes in das entsprechende ALPHA-Zeichen um und hängt es im ALPHA-Register an (APPEND).

2D (2 Bytes to Decimal): Ermittelt das dezimale Equivalent der letzten beiden Bytes eines ALPHA-Strings in X. Das dezimale Equivalent des vorletzten Bytes verbleibt in X; das des letzten Bytes kommt in M zu stehen.

OM (Open Memory): Erzeugt einen neuen Byte-String und tauscht den ursprünglichen Inhalt des c-Registers gegen diesen aus. Dadurch wird der Curtain (Grenze zwischen Daten- und Programm-Register) auf die Adresse 010 (hex) bzw. 16 (dezimal) gesetzt und die absolute Adresse des Σ-Registers auf 1FF (hex). Die Adresse des .END. wird nicht verändert. Dies erlaubt, Werte direkt in das Key-Assignment-Register und die Programm-Register abzuspeichern. Ein Programm kann während dieses Zustands ruhig gestoppt werden, ohne daß MEMORY LOST geschieht; globale Label sind weiterhin aufrufbar. Der in X stehende ursprüngliche Inhalt des c-Registers kann wie ALPHA-Werte abgespeichert und wieder abgerufen werden. (Warnung: Dieser Wert darf nicht verloren gehen, da er benötigt wird, um,durch das Rückspeichern ins c-Register, den Curtain wieder auf die alte Adresse zurückzubringen. Wenn OM allein für sich verwendet werden soll, muß irgendwo über diesem Programm ein END im Speicher stehen, damit nach der Ausführung dieser Routine man durch zweimal BST zu der X<>c-Instruktion zurückkommen kann, durch die der Curtain auf die alte Adresse zurück gebracht werden kann. OM ist eine von den sehr machtvollen Routinen, die Vorsicht und Sorgfalt in der Handhabung verlangen.)

VA (View Alpha): Zeigt den Inhalt des Alpha-Registers an, genau wie AVIEW. Wenn der Drucker angeschlossen und eingeschaltet ist und Flag 21 gesetzt ist, wird der Inhalt auch ausgedruckt. Ungleich zu AVIEW, hält VA das Programm niemals an.

XD (Hex to Decimal): Wandelt einen 2-Digit Hex-Wert (in Form von 2 Zeichen in Alpha) in eine dezimale Zahl um, die in X zu stehen kommt. Benutzt QR als Unterroutine.

QR (Quotient and Remainder): Ersetzt die Werte aus Y und X durch (Y - Y mod X)/X (der Quotient) und Y mod X (der Rest). Verändert nicht den Inhalt des Alpha-Registers, solange dieser nicht länger als 14 Zeichen ist.

LF (Locate Free Register): Überprüft das Key-Assignment-Register und gibt folgendes aus: X = aaa,eee, wobei aaa und eee die Adressen (relativ zur Curtain-Adresse 16 dezimal) sind des Anfangs und des Endes des Blocks freier Register zwischen dem Key-Assignment und dem .END. im Programm-Speicher; Y = c-Register-Inhalt für Curtain auf 16 (erzeugt durch OM). Benutzt OM und E? als Unterroutinen.

E? (.END. Finder): Ermittelt die absolute Adresse des Programm-Registers, welches das .END. enthält und plaziert die Adresse (in dezimal) in X. Benutzt 2D als Unterroutine.

Synthetische Instruktionen, die in LB, MK und den Unterroutinen verwendet werden.

Programm	Zeile	Synthetischer Kode (hex)
DC	74	Text: F5 1F FO 01 69 01
LB	80	3 Byte GTO 10: DO 00 0A
LB	111	TONE 87: 9F 87
LB	124	3 Byte GTO 03: DO 00 03
LB	164	3 Byte GTO 00: DO 00 00
LB	172	TONE 83: 9F 53
LB	173	3 Byte GTO 00: DO 00 00
LB	199	TONE 54: 9F 36
LB	219	3 Byte GTO: DO 00 08
LB	223	3 Byte GTO: DO 00 0A
MK	15	3 Byte GTO 07: DO 00 07
MK	62	3 Byte GTO 01: DO 00 01
MK	96	3 Byte GTO 08: DO 00 08
MK	109	3 Byte GTO 08: DO 00 08
MK	122	3 Byte GTO 09: DO 00 09
MK	135	Text: F3 2A 2A FO
MK	165	3 Byte GTO 03: DO 00 03
MK	178	TONE 83: 9F 53
MK	180	3 Byte GTO 01: DO 00 01
MK	193	TONE 83: 9F 53
MK	202	3 Byte GTO 01: DO 00 01
MK	212	Text: F8 2A 10 2A 00 00 2A 2A FO

....

Weitere Clubtreffen geplant !!

Auch in Siegen und Bremen sollen neue Ortsgruppen gegründet werden:

Siegen: Gerhard Schalk (481), Jägerstr. 9, Post Siegen 32, 5900 Niederschelderhütte
Tel. 0271 / 381829

Bremen: Jürgen Hüneke (419), Gerhard-Rohlfs-Str. 70, 2820 Bremen 7o
Tel. o421 / 650125

Interessenten melden sich bitte völlig unverbindlich zur Terminabsprache !!!!

....

Was bedeutet das ???

Zwei Mitglieder wunderten sich über seltsame Programmbefehle, z.B. auf Seite 263/4-81. Zur Klarstellung: steht im Programmlisting z.B. als "Befehl" F2 03 69 , so ist dies der Hexcode der jeweiligen Programmzeile.

....

Elektronik-Bastler gesucht !

Zum Erscheinungszeitpunkt dieses Infos haben wir hoffentlich bereits den ersten Osborne1 zur Verfügung. Gesucht wird ein Elektroniker, der von den Schnittstellen des Osborne (IEEE und RS, siehe im Osborne-Bericht) ein Interface zur Fotosatzmaschine von Rolf Hansmann, Limburger Str. 15, 6242 Kronberg 2 (M.Nr. 5oo) gegen Bezahlung erstellt. Wer Interesse verspürt, melde sich bitte bei Rolf !!!!!

Unterprogramm-Gruppe DC (Decimal to Character). Wird benötigt von der Programm-Gruppe LB (Load Bytes) und der Gruppe MK (Make Key Assignments). Erschienen im PPC Calculator Journal, Jahrgang 8 (1981), Heft 2 (März/April), Seite 34-38.

```
01◆LBL "DC"
02 INT
03 256
04 MOD
05 LASTX
06 +
07 OCT
08 X<> d
09 FS?C 11
10 SF 12
11 FS?C 10
12 SF 11
13 FS?C 09
14 SF 10
15 FS? 07
16 SF 09
17 FS? 06
18 SF 08
19 X<> d
20 X<> [
21 RCL \
22 "⊢*"
23 X<> ]
24 X<>Y
25 STO \
26 X<> ↑
27 "⊢*"
28 STO ↑
29 RDN
30 X<> ]
31 X<> \
32 STO [
33 RDN
34 RTN

35◆LBL "2D"
36 "*"
37 X<> [
38 X<> \
39 ASHF
40 "⊢◆↓◆▾◆◆
   "
41 X<> [
42 X<> \
43 X<> [
44 "⊢◆◆β"
45 RCL [
46 INT
47 +
48 RCL \
49 *
50 ST+ [
51 X<> \
52 RCL ]
53 INT
54 HMS
55 *
56 RCL ]
57 +
58  E1
59 ST* [
60 *
61 X<> [
62 RTN

63◆LBL "OM"
64 RCL c
65 STO [
66 "⊢◆◆◆◆◆◆"
67 X<> [
68 X<> d
69 CF 00
70 CF 01
71 CF 02
72 CF 03
73 X<> d
 74 "▓*i*"
75 X<> [
76 STO \
77 "⊢**"
78 X<> \
79 CLA
80 X<> c
81 RTN

82◆LBL "VA"
83 SF 25
84 PRA
85 SF 25
86 FS?C 21
87 CF 25
88 AVIEW
89 FC?C 25
90 SF 21
91 END
          CAT 1
LBL▾DC
LBL▾2D
LBL▾OM
LBL▾VA
END
200 BYTES
40 =
F77F00070060
0000
44 =
F47F000005
66 =
F67F00000000
00
74 =
F51FF0016901
```

Programm-Gruppe LB (Load Bytes). Bnötigt Unterprogramm-Gruppe DC (Decimal to Character). Erschienen im PPC Calculator Journal, Jahrgang 8 (1981), Heft 2 (März/April), Seite 34-39.
Blatt 1 von 2.

```
01♦LBL 00
02 STOP
03 GTO "++"

04♦LBL "LB"
05 FS? 50
06 GTO 00
07 "DEC/HEX
     INPT"
08 XEQ "VA"
09 CF 08
10 GTO 13

11♦LBL "L-"
12 CLA
13 XEQ "VA"
14 CF 08
15 RCL a
16 STO [
17 RCL b
18 FS? 08
19 GTO 14
20 STO \
21 SF 08

22♦LBL 13
23 RCL 11
24 CLST
25 STO 06
26 SIGN
27 ENTER↑
28 ENTER↑
29 R↑
30 GTO "++"

31♦LBL 00
32 RCL b
33 FC? 08
34 GTO 14
35 CLD
36 X<> [
37 STO a
38 X<> \
39 X<> b

40♦LBL 14
41 STO [
42 "⊢*****"
43 X<> \
44 XEQ "2D"
45 2
46 /
47 INT
48 LASTX
49 FRC
50 512
51 *
52 RCL [
53 +
54 7
55 *
56 +
57 115
58 X<>Y
59 -
60 7
61 XEQ "QR"
62 ST- Z
63 X<>Y
64 CHS
65 STO 09
66 X<> Z
67 LASTX
68 XEQ "QR"
69 1,001
70 ST* 09
71 ST+ Y
72 FRC
73 ST* T
74 X<>Y
75 R↑
76 +
77 *
78 X<>Y
79 X<=0?
80 GTO 10
81 ST+ 09
82 7
83 *
84 +
85 STO 07
86 XEQ "OM"
87 X<> c
88 STO 10
89 CLST

90♦LBL 06
91 STO 11
92 CLA

93♦LBL 07
94 ASTO 08
95 X<>Y
96 ISG 06

97♦LBL 15
98 SF 09
99 FS? 08
100 RTN
101 CF 22
102 CF 23
103 FIX 0
104 CF 29
105 "*"
106 ARCL 06
107 "⊢ OF "
108 ARCL 07
109 "⊢?"
110 XEQ "VA"
111 TONE 7
112 STOP
113 FS? 48
114 GTO 14
115 FC? 22
116 GTO 19
117 GTO 08

118♦LBL 14
119 FC? 23
120 GTO 19
121 XEQ "XD"

122♦LBL 08
123 X<0?
124 GTO 03
125 ENTER↑
126 CLA
127 ARCL 08
128 XEQ "DC"
129 RCL 06
130 X<=0?
131 GTO 10
132 7
133 MOD
134 X≠0?
135 GTO 07
136 X<>Y
137 RCL 09
138 RCL 10
139 X<> c
140 RCL [
141 STO IND
            Z
142 X<>Y
143 X<> c
144 R↑
145 RCL 08
146 DSE 09
147 GTO 06
148 ISG 09

149♦LBL 20
150 CF 09
151 CLX
152 X<>Y
153 RCL 07
154 FRC
155  E3
156 *
157 AOFF
158 FIX 0
```

Programm-Gruppe LB (Load Bytes). Benötigt Unterprogramm-Gruppe DC (Decimal to Character). Erschienen im PPC Calculator Journal, Jahrgang 8 (1981), Heft 2 (März/April), Seite 34-39.
Blatt 2 von 2.

```
159 "SST, DE
      L 00"
160 ARCL X
161 FIX 3
162 XEQ "VA"
163 BEEP
164 GTO 00

165♦LBL 01
166 RCL 06
167 X>0?
168 GTO 09

169♦LBL 10
170 "SST, MO
     RE +'S"
171 XEQ "VA"
172 TONE 3
173 GTO 00

174♦LBL 03
175 "*CORREC
      TION*"
176 XEQ "VA"
177 TONE 6
178 FC? 09
179 GTO 01
180 DSE 06
181 GTO 14
182 ISG 06
183 GTO 10

184♦LBL 14
185 RCL 06
186 7
187 MOD
188 X=0?
189 ISG 09
190 GTO 14
191 RCL 11
192 X=Y?
193 GTO 13
194 STO 08
195 RDN
196 STO 11
197 GTO 09

198♦LBL 13
199 TONE 4
200 6
201 ST- 06
202 R↑
203 GTO 15

204♦LBL 14
205 CLA
206 ARCL 08
207 ARCL 10
208 CLX
209 X<> \
210 STO [
211 ASTO 08

212♦LBL 09
213 RDN
214 GTO 15

215♦LBL "-B"
216 FC? 08
217 GTO 15
218 FS? 09
219 GTO 08

220♦LBL 19
221 RCL 06
222 X<=0?
223 GTO 10
224 CHS
225 ISG X
226 7
227 MOD
228 X=0?
229 GTO 14
230 CLA
231 ARCL 08

232♦LBL 11
233 "⊢♦"
234 DSE X
235 GTO 11
236 X<> [

237♦LBL 14
238 RCL 09
239 X<>Y
240 RCL 10
241 X<> c
242 X<>Y

243♦LBL 12
244 STO IND
           Z
245 CLX
246 DSE Z
247 GTO 12
248 RDN
249 X<> c
250 RDN
251 GTO 20

252♦LBL "XD"
253 "⊢♦Δ"
254 RCL [
255  E2
256 XEQ "QR"
257 29
258 ST- Z
259 -
260 ,9
261 ST* Z
262 *
263 INT
264 X<>Y
265 INT
266 16
267 *
268 +
269 RTN

270♦LBL "QR"
271 X<>Y
272 STO ]
273 X<>Y
274 MOD
275 ST- ]
276 LASTX
277 ST/ ]
278 CLX
279 X<> ]
280 X<>Y
281 END
         CAT 1
LBL'LB
LBL'L-
LBL'-B
LBL'XD
LBL'QR
END
558 BYTES
105 = <HEX>
F123
233 =<HEX
          CLA
233 = <HEX>
F27F00
253 = <HEX>
F37F0008
```

Programm-Gruppe MK (Make Key Assignments). Benötigt Unterprogamm-Gruppe DC (Decimal to Character). Erschienen im PPC Calculator Journal, Jahrgang 8 (1981), Heft 2 (März/April), Seite 34-40. Bàatt 1 von 2.

```
01◆LBL "MK"
02 CF 25
03 RCL 11
04 CF 07
05 SF 09

06◆LBL 01
07 XEQ "LF"
08 STO 09
09  E
10 +
11 X<>Y
12 STO 10
13 ASTO 11
14 DSE Y
15 GTO 07
16 SF 20
17 FC?C 09
18 GTO 13

19◆LBL 02
20 RCL 09
21 INT
22 LASTX
23 FRC
24  E3
25 *
26 X<>Y
27 ,5
28 FC? 10
29 SIGN
30 -
31 -
32 "REG FRE
       E: "
33 RCL d
34 FIX 1
35 ARCL Y
36 STO d
37 XEQ "VA"
38 TONE 6
39 PSE

40◆LBL 03
41 "PRE↑POS
    T↑KEY"
42 CLST
43 XEQ "VA"
44 TONE 7
45 STOP
46 GTO 14

47◆LBL "1K"
48 CF 20
49 SF 07

50◆LBL "+K"
51◆LBL 14
52 STO 08
53 RDN
54 STO 07
55 RDN
56 STO 06
57 CF 09
58 RCL 10
59 SIGN
60 FS? 20
61 X≠0?
62 GTO 01

63◆LBL 13
64 RCL 08
65 INT
66 X=0?
67 FS? 07
68 FC?C 20
69 GTO 02
70 X<0?
71 SF 09
72 ABS
73 STO Z
74 44
75 -
76 ABS
77 2
78 X<Y?
79 DSE T
80 R↑
81 STO Y
82  E1
83 ST/ Z
84 MOD
85 8
86 *
87 ENTER↑
88 CF 08
89 LASTX
90 FS? 09
91 ST+ Y
92 R↑
93 INT
94 X≠0?
95 X>Y?
96 GTO 08
97 R↑
98 +
99 ST+ Z
100 X<>Y
101 X<=Y?
102 CLX
103 X≠0?
104 SF 08
105 +
106 36
107 -
108 X>0?
109 GTO 08
110 FC? 09
111 RCL ▾
112 FS? 09
113 RCL e
114 FC? 08
115 GTO 14
116 STO [
117 "⊢*"
118 X<> [

119◆LBL 14
120 X<> d
121 FS? IND
           Y
122 GTO 09
123 SF IND Y
124 X<> d
125 FC? 08
126 GTO 14
127 STO [
128 ARCL 10
129 X<> \

130◆LBL 14
131 FC? 09
132 STO ▾
133 FS?C 09
134 STO e
135 "**"
136 FS? 10
137 ARCL 11
138 X<> Z
139 RCL 07
140 RCL 06
141 XEQ "DC"
142 XEQ "DC"
143 XEQ "DC"
144 FS?C 10
145 GTO 14
146 "⊢◆◆◆"
147 ASTO 11
148 SF 10

149◆LBL 14
150 RCL 09
151 RCL 10
152 X<> c
153 RCL [
154 STO IND
           Z
155 X<>Y
156 X<> c
```

Programm-Gruppe MK (Make Key Assignments). Benötigt Unterprogramm-Gruppe DC (Decimal to Character). Erschienen im PPC Calculator Journal, Jahrgang 8 (1981), Heft 2 (März/April), Seite 34-40.
Blatt 2 von 2.

```
157 CLST
158 CLA
159 FC? 10
160 ISG 09
161 SF 20
162 FS? 07
163 RTN
164 FS? 20
165 GTO 03
166 "DONE, N
     O MORE"
167 SF 09
168 GTO 14

169♦LBL 07
170 "NO ROOM
          "
171 CF 20
172 CLST

173♦LBL 14
174 FS?C 25
175 RTN
176 XEQ "VA"
177 TONE 7
178 TONE 3
179 STOP
180 GTO 01

181♦LBL 08
182 "NO SUCH
         KEY"
183 GTO 14

184♦LBL 09
185 X<> d
186 "KEY TAK
          EN"

187♦LBL 14
188 CF 09
189 CF 20
190 FS?C 25
191 RTN
192 XEQ "VA"
193 TONE 3
194 PSE
195 "KEYCODE
          ?"
196 CLST
197 RCL 08
198 XEQ "VA"
199 TONE 7
200 STOP
201 STO 08
202 GTO 01
```

```
203♦LBL "LF"
204 XEQ "E?"
205 17
206 -
207  E3
208 /
209 177
210 +
211 XEQ "OM"
212 "*8*♦♦**
           "
213 ,
214 ENTER↑
215 DSE T
216 GTO 14

217♦LBL 00
218 X<> IND
           T
219 X=Y?
220 GTO 14
221 X<> [
222 "⊢8"
223 STO \
224 ARCL X
225 RDN
226 RCL \
227 X<> IND
           T
228 ISG T
229 GTO 00

230♦LBL 14
231 X<> [
232 ARCL X
233 X<> \
234 SF 10
235 X=Y?
236 DSE T
237 CF 10
238 X<> Z
239 X<> c
240 R↑
241 RTN

242♦LBL "E?"
243 RCL c
244 XEQ "2D"
245 16
246 MOD
247 LASTX
248 X↑2
249 *
250 RCL [
251 +
252 CLA
253 END
```

```
       CAT 1
LBL"MK
LBL"1K
LBL"+K
LBL"LF
LBL"E?
END
535 BYTES
135 =
F32A2AF0
146 =
F47F000000
212 =
F72A102A0000
2A2A
```

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
8	128 ◆ DEG 00	129 ▪ RAD 01	130 x̄ GRAD 02	131 ← ENTER 03	132 α STOP 04	133 β RTN 05	134 Γ BEEP 06	135 ↓ CLA 07	136 Δ ASHF 08	137 σ PSE 09	138 ◆ CLRG 10	139 λ AOFF 11	140 μ AON 12	141 ∡ OFF 13	142 τ PROMPT 14	143 Φ ADV 15	8
9	144 Θ RCL 16	145 Ω STO 17	146 δ STO+ 18	147 Å STO- 19	148 å STO* 20	149 Ä STO/ 21	150 ä ISG 22	151 Ö DSE 23	152 ö VIEW 24	153 Ü ΣREG 25	154 ü ASTO 26	155 Æ ARCL 27	156 æ FIX 28	157 ≠ SCI 29	158 £ ENG 30	159 ▒ TONE 31	9
A	160 XROM 32	161 ! XROM 33	162 " XROM 34	163 # XROM 35	164 $ XROM 36	165 % XROM 37	166 & XROM 38	167 ' XROM 39	168 (SF 40	169) CF 41	170 * FS?C 42	171 + FC?C 43	172 , FS? 44	173 – FC? 45	174 . GTO IND XEQ IND 46	175 / SPARE 47	A
B	176 0 SPARE 48	177 1 GTO 00 49	178 2 GTO 01 50	179 3 GTO 02 51	180 4 GTO 03 52	181 5 GTO 04 53	182 6 GTO 05 54	183 7 GTO 06 55	184 8 GTO 07 56	185 9 GTO 08 57	186 : GTO 09 58	187 ; GTO 10 59	188 < GTO 11 60	189 = GTO 12 61	190 > GTO 13 62	191 ? GTO 14 63	B
C	192 @ GLOBAL 64	193 A GLOBAL 65	194 B GLOBAL 66	195 C GLOBAL 67	196 D GLOBAL 68	197 E GLOBAL 69	198 F GLOBAL 70	199 G GLOBAL 71	200 H GLOBAL 72	201 I GLOBAL 73	202 J GLOBAL 74	203 K GLOBAL 75	204 L GLOBAL 76	205 M GLOBAL 77	206 N X < > 78	207 O LBL 79	C
D	208 P GTO 80	209 Q GTO 81	210 R GTO 82	211 S GTO 83	212 T GTO 84	213 U GTO 85	214 V GTO 86	215 W GTO 87	216 X GTO 88	217 Y GTO 89	218 Z GTO 90	219 [GTO 91	220 \ GTO 92	221] GTO 93	222 ↑ GTO 94	223 _ GTO 95	D
E	224 ` XEQ 96	225 a XEQ 97	226 b XEQ 98	227 c XEQ 99	228 d XEQ 00 100	229 e XEQ 01 100	230 f XEQ A 102	231 g XEQ B 103	232 h XEQ C 104	233 i XEQ D 105	234 j XEQ E 106	235 k XEQ F 107	236 l XEQ G 108	237 m XEQ H 109	238 n XEQ I 110	239 o XEQ J 111	E
F	240 p TEXT 0 T	241 q TEXT 1 Z	242 r TEXT 2 Y	243 s TEXT 3 X	244 t TEXT 4 L	245 u TEXT 5 M [	246 v TEXT 6 N \	247 w TEXT 7 O]	249 x TEXT 8 P ↑	249 y TEXT 9 Q =	250 z TEXT 10 ⊦ ⊤	251 π TEXT 11 a	252 | TEXT 12 b	253 → TEXT 13 c	254 Σ TEXT 14 d	255 ⊢ TEXT 15 e	F
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

TABLE 2-1. THE HP-41C BYTE TABLE

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0	0 NULL 00	1 LBL 00 01	2 LBL 01 02	3 LBL 02 03	4 α LBL 03 04	5 LBL 04 05	6 Γ LBL 05 06	7 ↓ LBL 06 07	8 Δ LBL 07 08	9 σ LBL 08 09	10 ◆ LBL 09 10	11 λ LBL 10 11	12 μ LBL 11 12	13 ∡ LBL 12 13	14 LBL 13 14	15 LBL 14 15	0
1	16 0 16	17 Ω 1 17	18 2 18	19 3 19	20 ä 4 20	21 Ä 5 21	22 ö 6 22	23 Ö 7 23	24 ö 8 24	25 Ü 9 25	26 ü . 26	27 Æ EEX 27	28 æ CHS 28	29 ≠ GTO α 29	30 £ XEQ α 30	31 ■ SPARE 31	1
2	32 RCL 00 32 (space)	33 ! RCL 01 33	34 " RCL 02 34	35 # RCL 03 35	36 $ RCL 04 36	37 % RCL 05 37	38 & RCL 06 38	39 ' RCL 07 39	40 (RCL 08 40	41) RCL 09 41	42 * RCL 10 42	43 + RCL 11 43	44 , RCL 12 44	45 − RCL 13 45	46 . RCL 14 46	47 / RCL 15 47	2
3	48 0 STO 00 48	49 1 STO 01 49	50 2 STO 02 50	51 3 STO 03 51	52 4 STO 04 52	53 5 STO 05 53	54 6 STO 06 54	55 7 STO 07 55	56 8 STO 08 56	57 9 STO 09 57	58 : STO 10 58	59 ; STO 11 59	60 < STO 12 60	61 = STO 13 61	62 > STO 14 62	63 ? STO 15 63	3
4	64 @ + 64	65 A − 65	66 B * 66	67 C / 67	68 D X<Y? 68	69 E X>Y? 69	70 F X≤Y? 70	71 G Σ+ 71	72 H Σ- 72	73 I HMS+ 73	74 J HMS- 74	75 K MOD 75	76 L % 76	77 M %CH 77	78 N P-R 78	79 O R-P 79	4
5	80 P LN 80	81 Q X^2 81	82 R SQRT 82	83 S Y^X 83	84 T CHS 84	85 U e^X 85	86 V LOG 86	87 W 10^X 87	88 X e^X-1 88	89 Y SIN 89	90 Z COS 90	91 [TAN 91	92 \ ASIN 92	93] ACOS 93	94 ↑ ATAN 94	95 _ DEC 95	5
6	96 1/X 96 T	97 a ABS 97	98 b FACT 98	99 c X≠0? 99	100 d X>0? 00 100	101 e LN1+X 01 101	102 f X<0? A 102	103 X=0? B 103	104 h INT C 104	105 i FRC D 105	106 j D-R E 106	107 k R-D F 107	108 l HMS G 108	109 m HR H 109	110 n RND I 110	111 o OCT J 111	6
7	112 CLΣ T	113 X<>Y Z	114 r PI Y	115 s CLST X	116 t R↑ L	117 u RDN M	118 v LASTX N	119 w CLX O	120 x X=Y? P	121 y X≠Y? Q	122 z SIGN ⊢	123 X≤0? a	124 MEAN b	125 → SDEV c	126 Σ AVIEW d	127 ⊢ CLD e	7
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

HEWLETT PACKARD ANWENDER-CLUB
OLIVER RIETSCHEL

ACHTUNG : Das Loesungsbuch Nr. 4 Statik 1 ist fertig !

In dem Buch sind 17 Programme aus den Fachgebieten :
Stahlbetonbau , Holzbau , Stahlbau , allgemeine Statik und
Gewichtsermittlungen beschrieben.

Von den Clubmitgliedern
Claus Dachselt
Wolfgang Treutwein
Walter Pieperhoff
sind folgende Programme in das Loesungsbuch aufgenommen worden :
Symmetrischer Zweigelenkrahmen
Lastverteilung fuer vierseitig gelagerte Platten
Knicklaengen eines zweifeldrigen Druckstabes
Mehrteilig gespreizter Holzdruckstab
Einzellast auf kreuzweise gespannte Platte
Geduebelte Rahmenecke
Betonstahlgewichte
Lagermatten - Gewichte
Traegerauflager
Randspannung bei rechteckigen Querschnitten
Ein- u. zweireihiger , einfacher Winkelanschluss
Fundamentbalken fuer beliebig viele Einzellasten
2 Feldgerbertraeger
3 Feldgerbertraeger Gelenke i. d. Innenfeld
3 Feldgerbertraeger Gelenke i. d. Aussenfeld
Einzelfundament
Anschluss I - Profil an Stuetze (geschweisst)

Jedes Programm setzt sich zusammen aus
1. der Programmbeschreibung
2. einem Beispiel
3. dem Listing

BESTELLSCHEIN :
An : Walter Pieperhoff , Walb. Osth. Wallstr. 10 , 4770 Soest

Name : ______________________________

Anschrift : ______________________________

Ich bestelle :

______ mal Loesungsbuch Statik 1 73 Seiten 70,- DM ohne Magnetkarten

______ mal Loesungsbuch Statik 1 73 Seiten 100,- DM mit Magnetkarten

Preise inkl. Nebenkosten

______ Ich habe den Betrag auf das Konto Nr. 3522372 der Stadtsparkasse Soest BLZ 41450075 , ueberwiesen.

______ V. - Scheck liegt bei.

(zutreffendes bitte Ankreuzen)

Clubprojekt Nr. 1 (Magnetkarten,-Taschen,-Hüllen)

z. Hd.: Ulli Davertzhofen, Ostenbergstraße 97/8, 4600 Dortmund 50
Postscheckkonto-Nr. 2080 **27-432, PschA** Essen

Liebe Clubmitglieder,

Viele von Euch bitten bei ihrer Zubehör-Bestellung um "umgehende, **sofortige, dringende" Lieferung. Diese Wünsche** kann ich leider **nur selten erfüllen. Natürlich kommt es vor, daß** ich eine Bestellung **noch am Eingangstag erledigen kann,** allgemein sind gewisse **Wartezeiten jedoch unvermeidlich. Dies hat** vornehmlich zwei Gründe:

- **ich muß mit der Weiterleitung der Bestellungen** warten, bis die **Sammelbestellmengen erreicht sind. Andererseits** wären unsere **günstigen Preise nicht möglich.**
- **ich kaufe bei verschiedenen Händlern ein, so daß** es an der Regel **ist, daß ich die benötigte Ware** "scheibchenweise" geliefert **bekomme. Allerdings kann ich bei der** Auslieferung an Euch nicht **ebenso verfahren, da dies mit sehr** viel höheren Kosten verbunden wäre.

Ich bitte um Euer Verständnis, wenn einige Wochen vergehen, bis **Ihr das gewünschte Zubehör erhaltet. Sollten** sich einmal größere **Fristen ergeben (Z.B. 6 Wochen und mehr), so** werde ich die entsprechenden **Besteller kurz benachrichtigen, zumal** ich aus Kosten- **und** Zeitgründen **keine Eingangsbestätigungen** mehr verschicke und **man nach einigen Wochen verständlicherweise** zu grübeln beginnt, **ob Bestellung und Geld überhaupt angekommen sind.**
Schließlich wird Eure Geduld dadurch belohnt, daß Ihr Euer Zubehör **zu sehr günstigen Preisen bekommt.** Apropos: die Preise bleiben **konkurrenzlos, selbst wenn ich sie zum** 1.10.81 geringfügig **erhöhen muß(te).** Da mittlerweile der überwiegende Teil von Euch **auch DIN A4-Einlegeblätter bestellt,** wozu größere und teurere **Versandtaschen nötig sind, steigt** auch der Unkostenbeitrag um DM 0.50 .
Zum Abschluß: jetzt gibt es auch Tastenfeldschablonen zum Selbstbeschriften. **Einen Bestellzettel findet** Ihr unten.

An: Ulli Davertzhofen, Ostenbergstraße 97/8, 46 Dortmund 50

Name: ______________________ Mitgl.-Nr.: __________

Anschrift: ____________,( )______________

Ich bestelle:

_____ mal 40 Magnetleerkarten , je DM 19.00 = DM

_____ Stück Kartentaschen(Spiralheftung), je DM 9.50 = DM

_____ Stück Einlegeblätter(Klarsichtfolie,DIN A4, für je 15 Karten) , je DM 2.00 = DM

_____ Stück Tastenfeldschablonen , je DM 3.00 = DM

+ pauschaler Unkostenbeitrag(Verpackung,Porto) = DM 4.50

Gesamtbetrag = DM

( ) Ich habe den Betrag auf Pschk 2080 27-432 ,Essen überwiesen.

( ) Verrechnungsscheck liegt bei.

Lorenz Tichy 414
[illegible] Str. 51 Tel. 2 3786
8520 Erlangen

Erläuterungen zu "GLS"

Das vorliegende Programm dient dazu, n lineare Gleichungen mit n Unbekannten zu lösen. Die kooffizienten dieser sind in einer Matrix angeordnet (Abb.1). Das Programm dient nur der Lösung eines Gleichungssystems, auf Berechnung der Determinante und Inversen wurde verzichtet, um Speicherplatz zu sparen. Hierdurch unterscheidet es sich z.B. vom ähnlichen Programm "4x4" im Solution-Book: High level Math; dieses benötigt 102 (!) Programmregister und löst damit ausschließlich 4x4-Gleichungssysteme, während "GLS" je nach Anzahl der angeschlossenen Module bis zu 15x15-Matrizen lösen kann; es benötigt 45 Programmregister und n(n+1)+9 Datenregister. Die Rechenzeiten für n=2...11 sind aus Abb.2 ersichtlich.

$$\begin{matrix} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{matrix}$$

Abb.1

2	3	4	5	6	7
9"	20"	37"	1'03"	1'37"	2'21"
8	9	10	11		
3'17"	4'24"	5'50"	7'29"		

Abb.2

Algorithmus

Zur Anwendung kommt der sogenannte Gauß-Algorithmus mit Totalpivotisierung, der wahrscheinlich nicht allen von Euch bekannt ist. Dieser sei an einem Beispiel erläutert: Gegeben sind die drei Gleichungen $2x_1 + 3x_2 = 5$, $x_1 + x_2 + x_3 = 0$, $6x_1 + 7x_2 + 2x_3 = 7$, in Matrixform: (1). a_{11} ist das sogenannte Pivotelement, a_{21} soll durch Multiplikation der 1.Zeile mit einer gesuchten Zahl s und Addition der 1. zur 2. Zeile zu 0 gemacht werden, ebenso a_{31} (man darf ja Gleichungen addieren). Die gesuchten Zahlen sind -0,5 bzw. -3; daraus ergibt sich folgende Matrix (2). Da Zeilen (Gleichungen) mit beliebigen Zahlen muliplizíert werden dürfen, folgt Matrix (3). Jetzt wird a_{22} als Pivotelement gewählt und das Verfahren wiederholt; diesmal sollen a_{12} und a_{32} zu 0 gemacht werden (4), entsprechend folgt (5). Wenn nun die Diagonalelemente $a_{11} \dots a_{33}$ alle 1 und die übrigen Elemente der "homogenen" Matrix (des linken Teils, 3x3) 0 sind, stehen rechts von oben nach unten die Lösungen des Gleichungssystems, also ist $x_1 = -2$, $x_2 = 3$, $x_3 = -1$. Der Rechner geht im Prinzip genauso vor, er dividiert allerdings die Zeile, die das Pivotelement enthält, durch dieses, wodurch die Diagonalelemente alle 1 werden. Noch ein Hinweis, falls ein Pivotelement 0 ist: der Rechner geht dann auf Pivotsuche, d.h. er tauscht so lange Zeilen aus (Lbl Ø3), bis er ein Pivot ≠Ø gefunden hat. Falls das Gleichungssystem keine eindeutige Lösung hat, versagt das Programm.

$$\left(\begin{array}{ccc|c} 2 & 3 & 0 & 5 \\ 1 & 1 & 1 & 0 \\ 6 & 7 & 2 & 7 \end{array}\right) \quad (1)$$

$$\left(\begin{array}{ccc|c} 2 & 3 & 0 & 5 \\ 0 & -0,5 & 1 & -2,5 \\ 0 & -2 & 2 & -8 \end{array}\right) \quad (2)$$

$$\left(\begin{array}{ccc|c} 2 & 3 & 0 & 5 \\ 0 & 1 & -2 & 5 \\ 0 & 1 & -1 & 4 \end{array}\right) \quad (3)$$

$$\left(\begin{array}{ccc|c} 1 & 0 & 3 & -5 \\ 0 & 1 & -2 & 5 \\ 0 & 0 & 1 & -1 \end{array}\right) \quad (4)$$

$$\left(\begin{array}{cccc} 1 & 0 & 0 & -2 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & -1 \end{array}\right) \quad (5)$$

Mathe-Freaks sei deshalb empfohlen, sich nach der Ergebnisanzeige durch XEQ A die Matrix anzusehen, ob sie die Gestalt von (5) hat oder nicht. Anderen empfehle ich die Prüfroutine Lbl 14 . Diese wird an das Programm angehängt, zusätzlich werden folgende Befehle eingefügt: nach Zeile 149 XEQ 14 und nach Zeile 187 RCL Ø3 X<=Y? GTO 16 RUN .

Bedienung des Programms:

Tastenfolge	Anzeige	Kommentar
XEQ "GLS"	RANG?	
3 R/S	a1.1	Koeffizienten
2 R/S	a1.2	
Ø3 R/S	a1.3	
Ø R/S	b1	
5 R/S	a2.1	
...	...	
...	b3	
7 R/S	READY	
(XEQ A	a1.1=2,ØØØØ	Überprüfung der richtigen
	a1.2=3,ØØØØ	Werteingabe; während Anzei-
	...	ge (PSE) Wertänderung mög-
	READY)	lich.
R/S	X1=-2,ØØØØ	Ergebnisse
R/S	X2=3,ØØØØ	
R/S	X3=-1,ØØØØ	

Listing Lbl 14

Lbl 14 9 RCL Ø4 1E3 / + RCL Ø3 2 + 1E5 / + STO Ø1 1E-5 - RCL Ø3 + STO ØØ LBL 17 RCL IND ØØ * RCL IND Ø1 * ISG Ø1 CLD ISG ØØ GTO 17 X≠Ø? RTN LBL 16 "NO SOLVE" PROMPT .

Garantien für dieses Programm kann ich natürlich nicht übernehmen, aber über regen Zuspruch würde ich mich sehr freuen, vielleicht gelingt es jemand, das Programm zu kürzen?

Happy Programming!

Listing "GLS"

```
Ø1   LBL "GLS"
     CF ØØ
     CF 29
     "RANG?"
     PROMPT
     STO Ø3
     GTO Ø8
     LBL A
     SF ØØ
1Ø   LBL Ø8
     FIX Ø
     RCL Ø3
     1 E3
     /
     1
     +
     STO ØØ
     STO Ø1
     STO Ø2
2Ø   9
     STO Ø5
     RCL Ø3
     X/2
     LASTX
     +
     8
     +
     STO Ø4
     LBL Ø1
3Ø   "a"
     ARCL ØØ
     "├."
     ARCL Ø1
     XEQ ØØ
     ISG Ø1
     GTO Ø1
     "b"
     ARCL ØØ
     XEQ ØØ
4Ø   RCL Ø3
     ST- Ø1
     ISG ØØ
     GTO Ø1
     CF ØØ
     2
     +
     1 E5
     /
     9
5Ø   +
     STO ØØ
     "READY"
     PROMPT
     GTO Ø2
     LBL ØØ
56   FS? ØØ
     GTO Ø9
     PROMPT
     STO IND Ø5
6Ø   LBL 1Ø
     FIX Ø
     1
     ST+ Ø5
     RTN
     LBL Ø9
     "├="
     FIX 4
     RCL IND Ø5
     RND
7Ø   ARCL X
     AVIEW
     PSE
     STO IND Ø5
     GTO 1Ø
     LBL Ø2
     Ø
     STO Ø8
     RCL ØØ
     STO Ø5
8Ø   RCL Ø2
     INT
     8
     +
     RCL Ø3
     9
     +
     1 E3
     /
     +
9Ø   STO Ø1
     LBL 15
     RCL IND Ø5
     STO Ø6
     X=Ø?
     XEQ Ø3
     LBL Ø4
     RCL Ø5
     INT
     RCL Ø1
1ØØ  INT
     X≠Y?
     GTO Ø7
     RCL Ø3
     2
     +
     RCL Ø2
     INT
     -
     ST+ Ø1
11Ø  GTO Ø6
111  LBL Ø7
     RCL IND Ø1
     STO Ø7
     LBL Ø5
     RCL Ø6
     ST/ IND Ø5
     RCL IND Ø5
     RCL Ø7
     *
12Ø  ST- IND Ø1
     1
     ST+ Ø5
     ISG Ø1
     GTO Ø5
     1
     STO Ø6
     LBL Ø6
     RCL Ø3
     1
13Ø  +
     1 E3
     /
     RCL Ø2
     INT
     1
     -
     +
     ST+ Ø1
     RCL ØØ
14Ø  STO Ø5
     RCL Ø1
     RCL Ø4
     X>Y?
     GTO Ø4
     LBL 12
     ISG ØØ
     CLD
     ISG Ø2
     GTO Ø2
15Ø  RCL Ø3
     9
     +
     RCL Ø3
     1
     +
     1 E5
     /
     +
     STO Ø5
16Ø  RCL Ø2
     FRC
     1
     +
     STO Ø6
     LBL 13
```

```
166  "X"
     FIX Ø
     ARCL Ø6
     "⊢="
17Ø  FIX 4
     ARCL IND Ø5
     PROMPT
     ISG Ø5
     CLD
     ISG Ø6
     GTO 13
     STOP
     LBL Ø3
     RCL Ø5
18Ø  INT
     STO Ø6
     RCL Ø2
     INT
     RCL X
     RCL Ø8
     +
     RCL Ø3
     1
     +
19Ø  *
     8
     +
     +
     LASTX
     RCL Ø3
     1
     +
     +
     1·E3
2ØØ  /
     +
     STO Ø7
     LBL 11
     RCL IND Ø6
     X<> IND Ø7
     STO IND Ø6
     ISG Ø6
     CLD
     ISG Ø7
21Ø  GTO 11
     1
     ST+ Ø8
     GTO 15
214  END
```

Liebe Clubmitglieder!

Die Reaktion auf das Bubble-Sort Programm war zwar nicht ganz so groß wie bei SORT3, aber einige abdruckreife Programme habe ich doch erhalten.Herzlichen Dank!
Ein paar Seiten weiter werdet Ihr eine Optimierung des DEZ-HEX Programmes von prisma 216/81. Dort fehlt nach Zeile 11 ein CLA. Zum Schluß noch eine Frage:
WER kann mir das Barcode-Heft für die Programmsammlung "Höhere Mathematik" leihen?
Ansonsten viel Spaß mit den Programmen und schreibt mal wieder,

Euer Andreas

BUBBLE SORT

SIZE 000 !!!

EINGABE: AAA,EEE R/S.
WOBEI AAA=ANFANGSREG.
EEE=ENDREGISTER

DAS PROGRAMM VERWENDET
NUR STACK UND LAST X-REG
NACH DEM SORTIEREN STEHT
DIE HOECHSTE ZAHL IM
KLEINSTEN REGISTER.
BEI ALPHA-DATEN ER-
SCHEINT >ALPHA DATA<

© COPYRIGHT BY
H.Klemp 8/1981
6152

Happy
Programming
Ing.(grad.) Hagen Klemp (73)
Dianastr. 25
1000 Berlin 28

```
01•LBL "BSORT"
02 STO L
03•LBL 01
04 ENTER↑
05 ISG X
06 GTO 05
07 GTO 04
08•LBL 05
09 RCL IND Y
10 RCL IND Y
11 X<=Y?
12 GTO 03
13 X<> IND T
14 X<> IND Z
15 SF 00
16•LBL 03
17 R↑
18 ISG X
19 GTO 01
20•LBL 04
21 RCL L
22 FS?C 00
23 GTO 01
24 END
END          49 BYTES
```

Das Programm wird etwas schneller wenn man nach Zeile 24 ,0001 einfügt. Andreas

p.s.:Nähere Informationen zum Sortieren finden sich in CHIP 7/81



Programm: UNIVERSAL - BASISUMWANDLER

wandelt Zahlen aus einem beliebigen System in Zahlen eines anderen beliebigen Systems um.

Besondere Eigenschaften:
- beliebig viele Stellen und Nachkommastellen
- Genauigkeit der Nachkommastellen bis 10^{-8}
- beliebig große Basis (von der Rechnergenauigkeit begrenzt)
- hoher Ein - u. Ausgabekomfort
- typ. Rechenzeit:15s, max:40s

Anleitung:

Programm starten mit XEQ "BU"
Abfrage B1 / B2 beantworten mit Eingabe der Eingabebasis, ENTER, Ausgabebasis, R/S

"NR.?" Eingabe der umzuwandelnden Zahl.
Es kann unter 3 Eingabemodi gewählt werden:

1.) Eingabe ins X-Register, R/S. Anwendbar für alle Zahlen ohne Ziffern größer als 9
2.) Eingabe ins Alpha-Register. Auf ALPHA schalten, Zahl eintippen (bei Ziffern SHIFT nicht vergessen, die Buchstaben A-I können für 10-18 verwendet werden), R/S. Anwendbar für alle Zahlen ohne Ziffern größer als 18.
3.) Eingabe einzelner Digits in dezimalform ins X-Reg. Nach Eingabe jeder Stelle A drücken. Bei Nachkommastellen: vor der ersten Nachkommastelle SF 00 ausführen. Nach Beendigung der Eingabe: R/S Anwendbar für alle Zahlen.

Bei Modus 1 können max 10 Ziffern, bei Modus 2 max 14 Zeichen eingegeben werden.

Zwei Ausgabemodi stehen zur Verfügung:

1.) Normal - u. Hex-codierte Form im Alpha-Register für Ausgabesysteme mit Basis 2 - 19 . (Ausnahme: bei Ausgabebasis = 10 auch im X-Register) Maximal können 15 Stellen ausgegeben werden.
2.) Einzelne Digits werden in dezimalform durch Schrägstrich / getrennt über Alpha- Register ausgegeben, maximal 24 Zeichen.

Ausgabemodus 2 stellt sich automatisch ein, wenn
-- Eingabemodus 3 gewählt wurde
-- Ausgabebasis größer als 19.
Sollte nach Eingabe 3 Ausgabe 1 gewünscht werden, muß am Ende der Eingabe CF 03 ausgeführt werden.

Für den Fall, daß bei Ausgabe 2 die 24 Zeichen des Alpha-Registers nicht ausreichen, kann man Programmschritt 140 durch STOP ersetzen (oder ACX mit Drucker) und die Stellen einzeln auslesen.

Viel Erfolg,

Michael Fiedler
Am Fußgraben 12
6257 Hünfelden

Nach erfolgter Ausgabe, kann mit R/S zur nächsten Eingabe gegangen werden, wenn Ein u. Ausgabebasis gleich bleiben.

Anm.: einige synthetische Befehle werden verwendet.

```
01•LBL "BU"
02 CF 01
03 CF 02
04 CF 29
05 "B1 ↑ B2 ?"
06 PROMPT
07 STO 02
08 10
09 X=Y?
10 SF 02
11 RCL Z
12 STO 01
13 X=Y?
14 SF 01
15•LBL 04
16 CF 02
17 0
18 STO 03
19 -1
20 STO 00
21 CF 03
22 CF 23
23 "NR.?"
24 PROMPT
25 AOFF
26 FS? 23
27 GTO 00
28 FS? 01
29 STO 03
30 FS? 01
31 GTO 08
32 0
33 STO 00
34•LBL 07
35 RDN
36 STO Y
37 FIX IND 00
38 ISG 00
39 AOFF
40 RND
41 X≠Y?
42 GTO 07
43 CLA
44 ARCL X
45•LBL 00
46 1.01
47 STO ]
48 "├•"
49 2
50 RCL ]
51 X<Y?
52 GTO 00
53 -1
54 STO 00
55 GTO 01
56•LBL A
57 SF 03
58 GTO 09
59•LBL 02
60 1.01
61 STO ]
62 "├•"
63•LBL 01
64 21
65 RCL ]
66 LOG
67 RND
68 X>Y?
69 GTO 01
70 SF 00
71 GTO 03
72•LBL 01
73 30
74 -
75 9
76 X<>Y
77 X>Y?
78 DSE X
79•LBL 09
80 FS? 00
81 GTO 01
82 RCL 01
83 ST* 03
84 RDN
85 ST+ 03
86 GTO 03
87•LBL 01
88 RCL 01
89 RCL 00
90 Y↑X
91 *
92 ST+ 03
93 1
94 ST- 00
95•LBL 03
96 FS? 03
97 STOP
98 RCL \
99 SF 25
100 X≠0?
101 GTO 02
102•LBL 02
103 CF 25
104 CLA
105 19
106 RCL 02
107 X<Y?
108 SF 03
109 RCL 03
110 FS? 02
111 ARCL X
112 FS? 02
113 GTO 03
114 LN
115 RCL 02
116 LN
117 /
118 FIX 8
119 RND
120 INT
121 1
122 +
123 Y↑X
124 STO 00
125 FIX 0
126•LBL 05
127 RCL 00
128 1
129 X=Y?
130 "├."
131 RCL 02
132 ST/ 00
133 RCL 03
134 RCL 00
135 /
136 INT
137 FS? 03
138 ARCL X
139 FS? 03
140 "├/"
141 FS? 03
142 GTO 02
143 9
144 X<>Y
145 X>Y?
146 ISG X
147 AOFF
148 130
149 +
150 X<> [
151 RCL \
152 "├0"
153 1.1
154 STO \
155 "├0"
156 RDN
157 RCL \
158 CLA
159 RDN
160 STO \
161 RDN
162 STO [
163 ARCL Z
164•LBL 08
165 RCL 03
166 RCL 00
167 MOD
168 STO 03
169 1 E-8
170 X<=Y?
171 GTO 05
172•LBL 06
173 RCL 00
174 1
175 X<Y?
176 GTO 01
177•LBL 03
178 PROMPT
179 GTO 04
180•LBL 01
181 "├0"
182 FS? 03
183 "├/"
184 RCL 02
185 ST/ 00
186 GTO 06
187 END
```

```
LBL'BU
END          317 BYTES
.END.        04 BYTES
```

```
R00= "used   "
R01= "B in   "
R02= "B out  "
R03= "used   "
```

Lieber Osborne-Interessent !

Es gibt viele, viele Personalcomputer auf dem Markt. Bald zuviele, um eine eindeutige Kaufentscheidung zu treffen. Oft mit zuviel "Spielkram" und zu sehr beschränkten Möglichkeiten in der praktischen, professionellen Anwendung.
Hier will der weltbekannte Computer-Vater und Fachbuchautor Adam Osborne nun für Abhilfe sorgen. Mit dem OSBORNE1 ! Er übertrifft alle derzeit auf dem Markt befindlichen Personalcomputer in Leistung, Flexibilität und universellen Einsatzmöglichkeiten des Standardgerätes. Das Preis-/Leistungsverhältnis gar ist kaum noch vergleichbar. Doch wollen wir zunächst einmal Adam Osborne selbst zu seinem Produkt hören, das sich durch strikte Verwirklichung eines langersehnten, konsequenten Konzeptes auszeichnet:

"Wir brauchen keine noch leistungsfähigeren und damit teureren Mikrocomputer. Wir brauchen einen großen Preiseinbruch. Wir brauchen einen Mikrocomputer, der durch Nutzung bestehender Technologie auf einem wesentlich niedrigeren Preisniveau angesiedelt ist."

"Wir brauchen einen Mikrocomputer, der eine Vielzahl von Aufgaben zuverlässig erfüllt, ohne wie die derzeitigen Mikrocomputer durch eine Vielzahl notwendiger oder möglicher Peripherieeinheiten zu einem großen System anzuwachsen. Denn gerade das treibt die Kosten in die Höhe !"

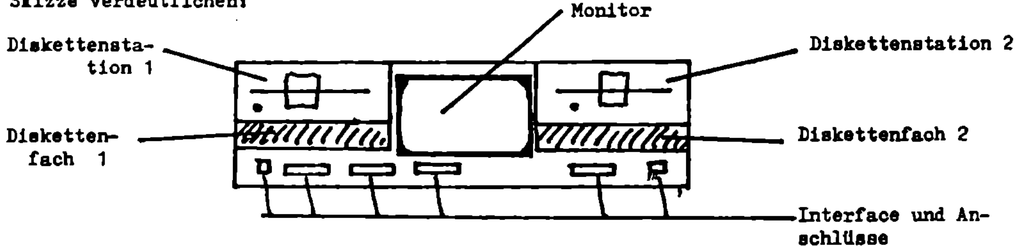
"Wir brauchen einen transportablen Mikrocomputer." - "Natürlich kann man auch einen Apple II in einen Stahlkoffer packen (ohne Monitor wohlgemerkt !) und in 15 bis 20 Minuten in Betrieb setzen. Warum keine aktenkoffergroßen, jederzeit betriebsbereiten Mikrocomputer ?"

"Wir brauchen einen Mikrocomputer, der hochwertige, professionelle Standardsoftware benutzt. Mit sehr vielen Anwendungsmöglichkeiten !"

"Der OSBORNE1 Mikrocomputer wurde geschaffen, um diese Anforderungen zu erfüllen. Und die Osborne-Computer-Gesellschaft wurde gegründet und finanziert, um den Absatz des Osborne1 zu unterstützen."

Soweit Adam Osborne, der geistige Vater des Osborne1.

Wie sieht dieser Traumrechner nun aus ? Er widerspricht allen bisherigen Design-Studien, sieht er doch aus, wie ein metallener Aktenkoffer: An der Oberseite ein Tragegriff - und sonst nur ein silbernes Metallgehäuse. Doch dann kommt die freudige Überraschung: Die Unterseite unseres "Koffers" können wir aufklappen und das Gerät vor uns auf den Tisch legen. Die abgenommene Unterseite stellt sich als Standard-ASCII-Tastatur mit abgesetztem numerischen 11er-Tastenfeld heraus. Insgesamt also 67 Tasten. Die Frontseite des Gerätes soll folgende Skizze verdeutlichen:



Standardmäßig verfügt der Osborne 1 über folgende HARDWARE:

- Z-80A CPU
- prof. Tastatur mit 11er-Block
- RS-232-C-Interface
- IEEE/488-vollimplement. Interface
- separates Modem-Interface
- Interface für externen Monitor
- 64 kByte RAM, davon 60 k für den Benutzer frei
- 2 Diskettenstationen 5.25 für je 102 kByte
- 2 Fächer für Diskettenaufbewahrung
- eingebauten 5"-Monitor mit 52x24 Zeichen, automatisch über 128 Spalten scrollend: 128 x 24 !!
- Anschluß für ext. Batterie/Akkusatz

Wer von dieser Komplettausstattung begeistert ist, soll sich erst einmal ansehen, was ohne Aufpreis als Standardsoftware mitgeliefert wird:

- CP/M Disk Operating System
- WordStar Textverarbeitungssystem der Spitzenklasse
- Mailmerge Adressverarbeitungssystem, ermöglicht u.a. auch Standard-/Rundschreiben mit WordStar

- Supercalc für dutzende kaufmännische und mathematische Anwendungen (entspricht Visicalc)
- MBASIC Strukturiertes Microsoft-BASIC - Interpreter
- CBASIC wie MBASIC - Compiler

Als weitere Hardware ist gegen geringen Mehrpreis erhältlich:

- 12" externer monitor 52 x 24 Zeichen
- 12" externer monitor 80 x 24 Zeichen
- Akustik-Koppler (z.B. für Telefonübertragungen oder Bandspeicherung)
- Batterie-/Akkusatz für bis etwa 5 Std. netzunabhängigen Betrieb
- Double-Density Diskettenstationen (doppelte Kapazität) - sehr empfehlenswert !
- Double-Density, Double-sided Diskettenstationen (vierfache Kapazität)

Im übrigen war der Osborne 1 bereits im deutschen Fernsehen zu bewundern: Im Bericht von de
Schachcomputerweltmeisterschaft, wo er für Aufsehen sorgte.
Und wann sorgt er in Deutschlands Wohnzimmern für Aufsehen ?? Ein erstes Demogerät erhalte
im November, offizieller Auslieferungsbeginn wird Anfang 1982 sein.

Und der PREIS ? Haben Sie auf 11.ooo,-- bis 13.ooo,-- DM geschätzt ? Das wäre heute marktüb-
lich ! Doch weit daneben: Der Listenpreis beträgt exakt 1795,-- US$. Für Deutschland kommen
Versicherung und Transport dazu, so daß man (1 US$ = 2,2o DM) noch unter 4500,--DM kommt !
Allein die Software kostet heute soviel - vereinzelt sogar noch mehr !

Außerdem werde ich zudem in der Lage sein, das Gerät unter Listenpreis zu liefern !! Natürl
ganz seriös und mit voller Garantie ! Bei Interesse bitte ich um Rückantwort mit nachstehen
Vordruck und einer 60 Pfg.-Briefmarke !

Mit freundlichen Grüßen

O. Rietschel

---

hier abtrennen und einsenden an:

Oliver Rietschel, Postfach 373, 2420 Eutin - bitte vorerst keine Telefonanrufe !

An dem Osborne 1 bin ich sehr interessiert ! Ich bitte daher darum,

( ) mir weitere Informationen sobald möglich zu übersenden, mit genauen Preisangaben, Lie-
ferzeiten, Fotos (liegen etwa zum Jahreswechsel vor)

( ) mich völlig unverbindlich als Kaufanwärter festzuhalten (Lieferzeitvorteil für Käufer,
Kalkulationsvorteil für Verkäufer, dadurch Preisvorteil für Käufer)

Vor- und Nachname: ______________________________

Adresse: ______________________________

Telefon: ____________________ Datum: ____________________

Für weitere Mitteilungen:

Hasso Schumacher (443)
Falenbek 61
2ooo Hamburg 73

10.10.1981

Lieber Oliver,
nachstehend, wie versprochen, ein Kurzbericht von meiner 3wöchigen USA-Reise, die mich nach New York und Houston, Texas, führte, teils privat, mehr aber geschäftlich. - In NY blieb ich fast ausschließlich nur in einem der 5 Boroughs (Stadtbezirke), nämlich Manhattan. In der 5th Avenue zB. gibt es allein Dutzende Geschäfte mit Radios, Koffern, Tapeten, Ansichtskarten und Taschenrechnern, immer in dieser für uns etwas seltsamen Zusammenstellung. Aber nur wenige Geschäfte dieser Art führen HP-Line, und wenn, dann nur den 41C(CV), selten Zubehör. Computer Shops gibt es ebenfalls sehr viele, die aber führen wieder kaum TR; am häufigsten fand ich 'Radio Shack's, fast immer mit dem Hinweis 'Computer Corner','Computer Shop' o.ä., welche ausschließlich den TRS in allen Versionen führten.- Also schlug ich nach in den 'Yellow Pages', dem Brachenadressbuch (es gibt deren 2: Privat und Commercial, die sind unterschiedlich umfangreich) nach Geschäften, die HP-Line führen. Ich wählte 2 aus: Manhattan Office Products Inc in der 43rd Street. Das Geschäft lag in der 21.Etage(!), hatte vollständiges 41C-System aufgebaut, auch zum Ausprobieren. Prospekte u.Preislisten lagen zur Mitnahme bereit. Gleich wurde ich gefragt, ob ich Hilfe benötigte. Auf meine Frage, ob sie ein Kassetten-Interface für den 41 hätten, holte er jedoch seinen Chef, der jedoch ebenfalls stirnrunzelnd den Kopf schüttelte und noch nie davon gehört haben wollte (Company policy ?!).Als ich nach vorhandenen Lösungsbüchern, Modulen u.ä. fragte, bedauerte man. Das andere Geschäft lag ganz im Süden von Manhattan, direkt hinter der Kirche am Trinity Place 95 (Super Business Machines Co.Inc), ein Ladengeschäft zu ebener Erde. Auch hier war man sehr hilfsbereit. Auf meine obligate Frage nach .. kam sofort die Antwort: In diesem Herbst noch erwarten sie eine Speichererweiterung von 16 k(!) und ein Kassetten-Interface! Von HP? Nein, von PPC. Na also. Er hatte davon gehört! PPC-Literatur ist ihm bekannt, aber sie ist nicht im Geschäft erhältlich. Aber alle Handbooks, Solution Books etc. waren da. Ebenso stand ein HP 85 da mit einem Demo Programm (eine Kopie anbei) und - brand new! - der neue HP 125. - HP selbst, am Pensylvania Plaza No 1 (dicht am Madison Square Garden) konnte ich erst nicht finden, also rief ich an: Sie haben dort nur ein Office, keinen Shop. Also unterließ ich einen Besuch dort.-In Houston - eine riesige Stadt von mehr als 60 km Durchmesser (!) fand ich in dem Teil der Stadt, in dem ich mich aufhielt (Down Town und Bell Air) wieder nur Tandy mit dem TRS. - Man sieht :es gibt drüben auch nicht mehr für den HP 41 als hier; ich hoffte eigentlich, auf dem US-Markt wäre schon mehr Zubehör zu finden als in 'Old Europe' . . .
Beste Grüße! Hasso

Vielen Dank für diesen Bericht, Hasso ! Vielleicht eifern Dir andere Mitglieder nach ihren Reisen einmal nach, so daß es in prisma nicht nur Prgr. zu "lesen" gibt!! Zu den SBM-Auskünften: Das gemeinte Kassetteninterface wird von Clubmitglied Helmut Jansen (283) gebaut. Ein PPC-Interface existiert meines Wissens kein eigenes. Die genannte Speichererweiterung ist das "Page-Switching", das Hans-Günther seit einem Jahr durchführt. Über beides wurde bereits ausführlich berichtet

Unter den von HP geplanten Peripherieeinheiten befindet sich meines Wissens weder eine 16k-Erweiterung für den 41c(v), noch ein Kassetteninterface. Auch in "Old Europe" sagt HP nicht mehr und nicht weniger als drüben .... nämlich fast nichts. Doch wenn neue Geräte kommen - in prisma wird es sofort zu lesen sein !

Oliver (1)

HEWLETT-PACKARD ANWENDER CLUB
OLIVER RIETSCHEL

ABTEILUNG : KARTENCOPYSERVICE

Ralf Pfeifer	Walter Piepenhoff
Rubensstr. 5	Walb. Osth. Wallstr. 10
5000 Koeln 50	4770 Soest
Tel. 0221/352034	Tel. 02921/2375

vorhandene Programme Liste Nr. 4	Anzahl d. Karten
133. LBL"ROM" springt absolute Adresse(i.00) im ROM an	1
134. LBL"SUB" ersetzt Xten Buchstaben i Alpha d. B. i Y	1
135. LBL"CS" speichert erz. Code aus X i.Reg.ind Y u.Y+1	1
136. LBL"CR" holt Code aus Reg. ind. Y u. Y+1 nach X	1
aus der Standardprogrammsammlung :	
137. UPN Lehrgang	2
138. Kalenderrechnung	2
139. Wortratespiel	2
140. Arithmetik - Lehrgang	2
141. Hexadezimal - Dezimal - Umwandlung	2
142. Finanztechnische Berechnung	2
143. Nullstellenbestimmung	1
144. Kurvenanpassung	2
145. Vektor - Operationen	1
146. Black - Jack (17 + 4)	3
aus Prisma 4/1981	
147. Texteingabe Hans - Guenter Goetter (S.202/81)	1
148. UEBE PRGM (verb. Arithm. Lehrgang) (S.204/81)	1
149. NEWTON - Verfahren (Mathematik) (S.206/81)	1
150. Lineare Regression m.Fehlererk. (Mathem.)(S.207/81)	1
151. Primfaktorzerlegung Hauke Trentens (S.208/81)	1
152. Primfaktorzerlegung Juergen Moeck (S.209/81)	1
153. Load Bytes (Laed Dezim. Codes i.d. Prgsp.)(S/215/81)	3
154. Differentialgleichung (Mathem.) (S.221/81)	3
155. Faktor - Finder (Mathem.) (S.234/81)	2
156. Plotten v. 2 Funktionen (S.245/81)	1
157. Balkendiagrammplotter (S.247/81)	2
158. Byte Print (Druckt Barcodes) (S.251/81)	1
159. Bar Alpha (Druckt Alpha - Data - Codes) (S.252/81)	1
160. ORDO (Spiel) (S.	2
161. Hindernisfahrt (Spiel) (S.263/81)	3
161. Tower of Hanoi (Spiel) (S.266/81)	1
162. MK Multiple Key Assignment (Synth.) (S.226/81)	4
163. LB Byte - Loading - Programm (Synth.) (S.225/81)	4
164. Load Bytes (S.216/81) (Synth.)	3
165. 128 Toene (Statuskarten) (Synth.)	4
166. Vektorrechnung (S.278/81) (Mathem.)	3
167. Gleichungen 4. Grades (S.278/81) (Mathem.)	2
168. Primzahlenprogramm (S.287/81) (Mathem.)	1
169. Star Treck (Spiel) (S.288/81)	2
170. Pruefziffernprogramm verbessert (Mathem.) (S.290/81)	1
171. Pruefen logischer Gesetze (S.292/81)	4
172. Automobilkosten (S.296/81)	5
173. AOS fuer UPN - Rechner (S.300/81)	2

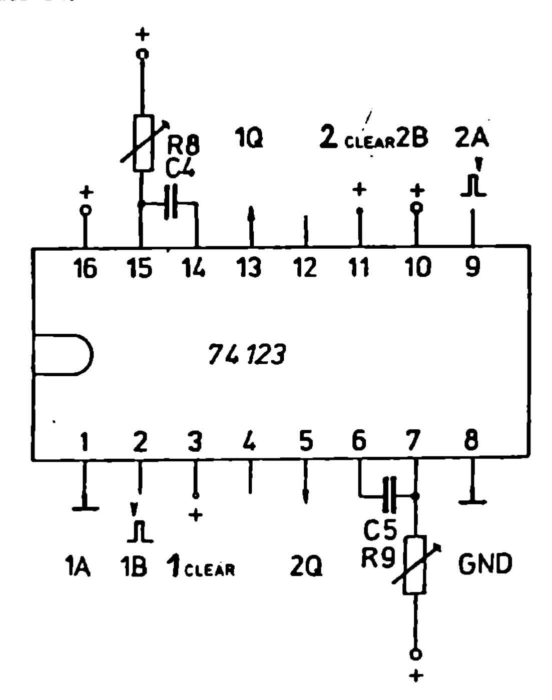
Bei Bestellungen : bitte nicht nur die Nr. des Programms sondern auch den Namen angeben . Erfoderliche Kartenanzahl und frankierten ; adressierten Briefumschlag zuschicken.

Beachtet unbedingt die Ubernächste Seite (405)

K A S S E T T E N - I N T E R F A C E

Ich habe zwischenzeitlich am Kassetten-Interface weitergearbeitet und dabei die folgenden Punkte herausgefunden, die helfen können, die Zuverlässigkeit des Interfaces sehr zu steigern. Da diese Verbesserungen in frühen Ausgaben des Handbuches nicht erwähnt sind beziehungsweise in frühen Interfaces nicht implementiert sind, seien sie hier zum selber-nachrüsten aufgeführt. Es besteht aber auch die Möglichkeit, mir Interfaces zuzuschicken; ich werde sie dann kostenlos auf den nachfolgend geschilderten, neuesten technischen Stand bringen. Voraussetzung dafür ist, daß am Gerät - und hier insbesondere an der Pin-Belegung - nichts verändert worden ist. Nach Mitte September gelieferte Geräte beinhalten die hier geschilderten Verbesserungen bereits von Hause aus.

- Die untenstehende Zeichnung zeigt die Verdrahtungsanweisung für das IC "74123" (oder "74LS123"). Die beiden Verbindungsleitungen zwischen Pin 3 und der positiven Versorgungsspannung sowie zwischen Pin 11 und der positiven Versorgungsspannung sollten nachträglich eingelötet werden.



o Die Praxis hat gezeigt, daß die Tonbandqualität einen erheblichen Einfluß auf die Anzahl der Fehllesungen des Kassetten-Interfaces hat. Es sollte deshalb nur erstklassiges Bandmaterial verwendet werden. Sehr gut bewährt hat sich zum Beispiel eine Tonbandkassette der Marke "Maxell UD 60". Sorgfältige Messungen haben gezeigt, daß dieses Bandmaterial entschieden weniger "Drop-Outs" erzeugt, als die großen Marktführer und die Billigfabrikate.

o Es hat sich herausgestellt, daß die beiden Spindeltrimmer wesentlich weniger empfindlich auf Verstellen reagieren als erwartet. Es kann deshalb folgende Empfehlung gegeben werden: Sollte Ihre Interface-Tonbandgeräte-Kombination gegen Fehllesungen anfällig sein (was entweder auf ein 'Altern' des Interface-Oszillators oder auf sehr schlechten Gleichlauf des verwendeten Bandgerätes zurückzuführen ist) , so sollten Sie die beiden Spindelpotentiometer etwas in Richtung auf höheren Widerstandswert (siehe hierzu auch obenstehende Abbildung) verstellen. Dadurch wird zwar die maximal übertragbare Baudrate (und also auch die maximale Abtastgeschwindigkeit von Barcode) leicht abgesenkt, andererseits aber die Sicherheit gegen Fehllesungen gesteigert.

o Ein vierter Punkt sei der Vollständigkeit halber vermerkt; er hat keinen Einfluß auf das Funktionieren des Interfaces, gehört aber zum 'Guten Stil der Digitalen Schaltungstechnik':
Alle nicht benutzten Eingänge des ICs "7400" (oder "74LS00") und des ICs "7408" (oder "74LS08" oder "74H08") sind mit einem festen TTL-Potential - also entweder mit der positiven Versorgungsspannung oder mit Masse - zu verbinden.

Ulrich Jansen (263)

__Magnetkartencopyservice !__

Mit dieser so lobenswerten Einrichtung unseres Clubs wird zunehmend Mißbrauch getrieben. Täglich gehen allein bei Walter durchschnittlich zwei Briefe mit 40 bis 100 Magnetkarten ein und der Bitte: Einmal alle Programme bitte ! Resultat: Ralf und Walter haben täglich mind. eine Stunde zu tun und der erste Kartenleser ist verheizt. 500,--DM, für die keiner geradesteht - Privatschaden. Daher müssen wir Gegenmaßnahmen ergreifen: Zusätzlich zu dem selbstadressierten, frankierten Rückumschlag und den Magnetkarten sind beizufügen:

__50 PFG! JE MAGNETKARTE__ zum Ausgleich des entstandenen und entstehenden Schadens !!

Zuschriften, denen dieser Betrag fehlt, werden mit unbeschrifteten Karten zurückgeschickt. Damit aber nicht für jedes kleine Programm ein oder zwei Mark bezahlt werden müssen und weil an diesem Zustand nur die Großbesteller schuld sind, gilt folgende Regelung:

Jedes Quartal hat jedes Mitglied ein Programm beliebiger Länge frei ! (also 4 p.a.)

Die Einhaltung dieser Regel wird genauestens überprüft. Daher und zur gleichmäßigen Arbeitsverteilung wenden sich bitte Mitglieder mit ungerader Mitgliedsnummer nur an Walter, solche mit gerader Mitgliedsnummer nur an Ralf.

So bedauerlich diese Regelung ist, ich muß um strikte Einhaltung bitten !

Oliver (1) i.A. des Copyservices

Diverse astronomische Programme

für HP-41C: SIZE 010, 847 Bytes = 121 Prgm-Speicher (= + 2 RAMs), 411 Prgm-Schritte. USER an, Drucker auf MAN (arbeitet auch ohne Drucker); der Zeitzähler sollte nur ohne Drucker laufen. SF 28, CF 29; vor dem Aufzeichnen des Prgm auf Magnetkarte Flag 11 setzen.

Bei allen Programmen außer Sternzeitzähler wird die Eingabe rechtsbündig ohne Kennzeichnung, die Werteausgabe linksbündig mit Kennzeichnung gedruckt.

§ Programme, für die keine Zeiteingabe nötig ist.
¶ Programme, die weder Datum- noch Zeiteingabe erfordern.

Für alle nicht gekennzeichneten Programme enthält die Datum- (= Start-)routine Konstanten, die vor Rechenbeginn auch dann abgespeichert werden müssen, wenn keine Berechnungen (z.B. beim Eichen des Zählers) durchgeführt werden.

Bei Berechnungen im Äquator- oder Polbereich können sich Schwierigkeiten (z.B. bei einer Division durch cos 90° = 0) ergeben. In diesen Fällen wird die Rechenroutine mit Flag 25 zu Ende geführt, damit die gesetzten Flags zuverlässig gelöscht werden, und am Schluß der Berechnung "FEHLER" angezeigt. Die im X- und ggfs. im Y-Register stehenden Ergebnisse sind dann nicht brauchbar. Die Schiefe der Ekliptik stimmt mit den im "Explanatory Supplement to the American Ephemeris" gegebenen Werten und Bespielen überein, nicht aber mit den Werten im Astronomical Almanac (Abweichungen um wenige Sekunden; warum, ist unbekannt).

Das Prgm arbeitet vom 1. 03. 1900 bis zum 31. 12. 1999. Der Nullpunkt der Kalenderroutine ist der 29. Oktober 1899. Für Berechnungen ab 1. 01. 2000 müssen die Konstanten auf den 29. 10. 1899 umgerechnet werden.

Die Ergebnisse stehen in der Reihenfolge im Stack (1. Ergebnis im Y-Register, 2. Ergebnis im X-Register), daß ggfs. ohne Neueingabe damit weitergerechnet werden kann. Sie werden als °.'" bzw. als h.ms mit folgenden Kennzeichnungen ausgegeben:

REKT = α = Rektaszension (h.ms)	LNGE = λ = ekliptikale Länge
DEKL = δ = Deklination	BRTE = β = ekliptikale Breite
HOHE = h = Höhe über Horizont	STD∢ = t = Stundenwinkel
AZIM = Az = Azimut	STZT = θ = Sternzeit
ZDIS = z = Zenitdistanz	POS∢ = PW = Positionswinkel
∢HHE = Winkelhöhe	∢ENT = Winkelentfernung

Eingaben mit ↑: mit ENTER↑ eingeben; α als h.ms 0 bis 24 Stunden, alle anderen Werte als °.'" 0° bis 360° bzw. 0° bis 90° eingeben, lediglich bei B 6 werden die Eingaben im Längenmaß gefordert.

	Taste	XEQ
I. Datum für alle nicht mit ¶ gekennzeichneten Programme eingeben ...	Σ+	DIV
Monat und Jahr zweistellig (Jahr ohne Jahrhundert), also z.B. 15. Januar 1982 = 15.0182. Nach der Frage "ZEIT?"		
II. für alle nicht mit § oder ¶ gekennzeichneten Programme die Uhrzeit (für Sternzeit-Zähler einige Minuten Vorbereitung einplanen) eingeben und	R/S	
Angezeigt wird die Sternzeit θ. Bei Sommerzeit vor der Eingabe eine Stunde abziehen.		
Die Uhrzeit kann jederzeit neu eingegeben werden mit	Σ-	ZEIT

A Koordinaten-Umwandlungen:

	Taste	XEQ
1. Äquator- in Horizontkoordinaten umrechnen: h und Az (von Süd über West) berechnen; eingeben (α=h.ms) ... α↑δ	$^1/_x$	EQ↑HOR
2. Horizont- in Äquatorkoordinaten umrechnen: α und δ berechnen; eingeben (Az Süd über West) ... ... Az↑h	y^x	HOR↑EQ
3.§ Äquator- in Ekliptikkoordinaten umrechnen: λ und β berechnen; eingeben (α=h.ms) . ... α↑δ	$\sqrt{x}$	EQ↑EKL
4.§ Ekliptik- in Äquatorkoordinaten umrechnen: α und δ berechnen; eingeben .. .. λ↑β	x^2	EKL↑EQ

B VERSCHIEDE[illegible] PROGRAMME:

	Taste	[illegible]
1. Zenitdistanz z berechnen; eingeben (α = h.ms): ... ... $\alpha\uparrow\delta$	10^x	ZDI
2. Stundenwinkel t berechnen; eingeben (α = h.ms): .. .. α	LOG	SA
3.§ Gestirnshöhen: Für Auf- bzw. Untergang ist für die Gestirnshöhe h nicht O, sondern wegen der Refraktion -.3 zu setzen, bei Sonne und Mond wegen der Größe der Gestirne -.51. Höhen unter dem Horizont sind negativ. "FEHLER" zeigt an, daß die gefragte Höhe nicht erreicht werden kann.		
a) Zeitpunkt für den Aufstieg auf eine bestimmte Höhe h über dem Horizont; Eingabe (α = h.ms) .. .. $\alpha\uparrow\delta$	LN	T↑
nach Aufforderung "HOEHE?" die Gestirnshöhe h eingeben ... h	R/S	
b) Zeitpunkt für das Absinken auf eine bestimmte Höhe über dem Horizont; Eingabe (α = h.ms) .. .. $\alpha\uparrow\delta$	e^x	TA
nach Aufforderung "HOEHE?" die Gestirnshöhe h eingeben ... h	R/S	
4.§ Kulminationszeit und -Höhe eines Gestirns; Eingabe (α = h.ms). $\alpha\uparrow\delta$	R↓	KU
5.¶ Positionswinkel (von Nord über Ost) und Winkelentfernung zweier Sterne voneinander berechnen: zuerst die Koordinaten des ersten Gestirns eingeben . ... $\alpha_1\uparrow\delta_1$	%	PA
Nach der Aufforderung "2. EINGABE" die Koordinaten des zweiten Gestirns eingeben ... ... $\alpha_2\uparrow\delta_2$	R/S	
6.¶ Höhe des tatsächlichen Horizonts über dem mathematischen Horizont in Winkelmaß berechnen; Eingabe . ... $\Delta h\uparrow E$	SIN	HO
Δh = Höhenunterschied zwischen Beobachtungsort und dem die Sicht begrenzenden Geländepunkt, E = Luftlinienentfernung zwischen diesen beiden Punkten (eine topografische Karte leistet gute Hilfe). Die Erdkrümmung ist wegen der wohl meist im Verhältnis geringen Entfernungen nicht berücksichtigt. Auf gleiche Maßeinheiten achten!		

C STERNZEIT-ZÄHLER:

1. Zähler: Der Drucker sollte nicht angeschlossen sein, da der Zeitzuwachs auf Rechner ohne Drucker abgestimmt ist. Bei angeschlossenem, aber abgeschalteten Drucker arbeitet der Zähler wegen der "VIEW"-Anzeige ohnehin nicht.		
1.1 Zähler einschalten	x≶y	ZH
In der Anzeige steht "AN" und Flag 1 ist gesetzt.		
1.2 Den Zähler starten mit	XEQ 04	
und im "Leerlauf" mindestens einmal durchlaufen lassen, damit das Prgm den Weg GTO 20 → LBL 20 speichern kann.		
Den Zähler wieder anhalten mit	R/S	
1.3 Datum (wenn nicht schon eingegeben; ggfs. s. Schritt I) und Startzeit eingeben (einige Minuten Vorbereitung einplanen)	Σ-	ZE
Nach Anzeige der Sternzeit θ die Rektaszension α als h.ms eingeben ... ... α	LOG	SA
Vor der Zählroutine hält der Rechner an. Nun R/S oder SST drücken, bis erst "GTO 04" und dann "NULL" zu sehen ist. Wenn die eingegebene Startzeit erreicht ist, die Zählroutine starten mit	R/S	
Da die Anzeige Sternzeit ist, stimmt sie nicht mit der Uhrzeit überein!		
1.4 Zu gegebener Zeit den Zähler stoppen mit	R/S	
1.5 Zähler wieder ausschalten	x≶y	Z
Mit dieser Routine kann der Zähler beliebig ein- und ausgeschaltet werden; entsprechend steht in der Anzeige "AN" und Flag 01 ist gesetzt bzw. in der Anzeige "AUS" und Flag 01 ist gelöscht.		

C Sternzeit-Zähler (Fortsetzung)

	Taste	XEQ
2.§ Eichen des Zählers. Hierzu muß der Zähler nicht eingeschaltet sein, doch die Datum-Routine muß zuvor zum Abspeichern von Konstanten gelaufen sein (wenn auch nicht unbedingt mit Datum-Eingabe).		
2.1 Start	CLΣ	EICH
Zur Kontrolle ist Flag 4 gesetzt		
2.2 Den Zähler starten mit	XEQ 05	
und mindestens einmal durchlaufen lassen (s. Schritt 1.2)		
Den Zähler wieder anhalten mit ...	R/S	
2.3 Zur Zählroutine gehen mit	GTO 05	
R/S oder SST drücken, bis erst "LBL 05" und dann "NULL" erscheint.		
Dann die Zählroutine starten mit .	R/S	
und gleichzeitig eine normale (nicht Sternzeit-) Stoppuhr starten (ein Start der Zählroutine mit XEQ 05 wäre gegen die Stoppuhr zu langsam).		
2.4 Einige Minuten nach dem Start die Zählroutine stoppen mit	R/S	
und gleichzeitig die Stoppuhr anhalten. Sollte 0.0003 oder 0.0001 in der Anzeige stehen, zweimal SST drükken.		
2.5 Die Stoppuhrzeit in den Rechner eingeben (die vom Zähler gemessene Zeit steht noch in der Anzeige) und	CLΣ	EICH
Flag 4 ist gelöscht. Damit ist die Eichung des Datenspeichers beendet.		

2.6 Für eine Kontrolleichung wiederholt man die Schritte 2.1 mit 2.5. Nur Schritt 2.2 kann ausgelassen werden, wenn nicht inzwischen auf PRGM geschaltet oder PACK ausgeführt wurde.

2.7 Zur Eichung des Programm-Speichers notiert man den Zeitzuwachs, der in der Anzeige steht, und geht im PRGM-Schaltung zu Zeile 049 (GTO.049), löscht den alten Wert (303583 E-9) und gibt die neue Zahl ein. Nach PACK und dem Neubelesen der Magnetkarten (mit gesetztem Flag 11) ist die Eichung beendet.

2.8 Nach jedem Eingriff in das Programm, der mit einer Änderung des Abstandes LBL 04 → LBL 20 verbunden ist, muß neu geeicht werden.

D Änderungen für Ortswechsel:

Die ortsspezifischen Konstanten (geografische Länge und Breite) sind für den jeweiligen Benutzerort einzugeben, und zwar

λ in h.ms bei Prgm-Zeilen 266 und 337
φ in °.'" bei Prgm-Zeile 074, λ_z Zeile 274, 329.

Im vorliegenden Programm sind die Werte für Neuburg an der Donau gespeichert: λ = -0 h 44 m 43 s (= -11° 10' 45"), φ = 48° 44' 20".

Beispiele: 17. Oktober 1981 (= 17.1081), 21.48 Uhr für Neuburg (Donau): λ = 11° 10' 45", φ = 48° 44' 20"; θ = 23h 17m 47s

	α 1950	δ 1950	Az	h	λ	β	ZDIS z	STDÄ t	T↑[1]	TAB[1]	KULM	KULM
	h m s	° ' "	° ' "	° ' "	° ' "	° ' "	° ' "	h m s	h m s	h m s	h m s	° ' "
M 31	0 40 00	41 00 00	290 24 01	73 34 42	27 09 16	33 21 06	49 00 00	22 37 47	21 48 00	0 31 59	23 10 00	82 15 40
β And	1 06 56	35 21 22	293 35 03	65 55 19	29 42 28	25 56 36	24 04 41	22 10 51	21 48 00	1 29 38	23 36 51	76 37 02
Sirius[2]	6 44 57	-16 38 46	264 57 39	-26 44 14	104 00 42	-39 32 34	116 44 14	16 32 50	21 48 00	12.43 49	5.17 53	24 36 54
Wega	18 35 15	38 44 09	106 31 01	39 51 48	284 36 54	61 43 52	50 08 12	4 42 32	12 24 29	21 48 00	17 06 14	79 59 49

	PÄ ° ' "	ÄENT ° ' "		Beob.-Ort Höhe ü.NN	Horiz-Pkt Höhe ü.NN	=Δh	Luftlin.	ÄHHE ° ' "
M 31 → β And	129 47 38	8 46 39		410 m	530 m	120 m	1500 m	4 32 26
Wega → Ring-Nebel	144 28 48	7 04 34		1020 m	2000 m	980 m	1800 m	28 33 57

[1] Wenn Az > 180°, entspricht die Aufstiegszeit der vorgegebenen Zeit, andernfalls die Abstiegszeit.
[2] Nicht sichtbar

FORMELN:

GST = Greenwich-Sternzeit um 0.00 Uhr UT am gegebenen Tag
GST_{1899} = Grennwich-Sternzeit am Kalender-Nullpunkt 0.00 Uhr = 2,473 489 169 h
d_o = Tage seit dem Nullpunkt der Kalenderroutine = seit 29. 10. 1899
ε = Schiefe der Ekliptik am gegebenen Tag
ε_{1899} = Schiefe der Ekliptik am Kalender-Nullpunkt = $23^\circ\!,452\ 317\ 25$
φ = geografische Breite (südliche Breiten negativ)
λ = geografische Länge (östlich von Greenwich negativ)
λ_z = Länge des Zeitzonenmeridians (östlich von Greenwich negativ) = λ h.ms auf Ganze gerundet.
R = Verhältnis Sonnentag/Sterntag = $24^h : 23^h\ 56^m\ 4^s\!,1 = 1{,}002\ 737\ 909$

d = Tag, m = Monat, j = Jahr, T = Uhrzeit, θ = Sternzeit
Gz = ganzzahliger, nicht gerundeter Teil einer Dezimalzahl (Vorkommateil)
Bt = Bruchteil einer Dezimalzahl (Nachkommateil)
h.ms = Angabe in Stunden, Minuten, Sekunden
°.'" = Angabe in Grad, Bogenminuten, Bogensekunden

Der Nullpunkt der Kalenderroutine ist der 29. 10. 1899; die Routine arbeitet mit den gegebenen Konstanten vom 1. 03. 1900 bis zum 31. 12. 1999. Bei Ortswechsel sind die ortsspezifischen Konstanten (φ, λ*) lt. Punkt 5 der Gebrauchsanweisung zu ändern. Das vorliegende Programm enthält die Konstanten von Neuburg an der Donau.

*und λ_t

....

Zeit- und Winkelroutinen:

$D_o = \frac{d_o}{36525}$; $T = \frac{\theta + \lambda - GST}{R} - \lambda_z$; Sternzeit $\theta = GST + (T + \lambda_z)R - \lambda$;

Stundenwinkel $t = \theta - \alpha$; $\alpha = \theta - t$; $\theta = \alpha + t$;

$GST = 2^h\!,473\ 489\ 169 + 2400{,}051\ 262 \cdot D_o + 25\ 806 \cdot D_o^2 \cdot 10^{-9}$

$\varepsilon = 23^\circ\!,452\ 317\ 25 - 0{,}013\ 012\ 5 \cdot D_o$; < 24 Stunden $= 24 \cdot \mathrm{Bt}\ \frac{24 + t}{24}$;

....

LBL DIV (Kalenderroutine):

$d_o = \mathrm{Gz}\,(365{,}25j') + \mathrm{Gz}\,(30{,}6m') + d$; m.jj − 11,99 = m + 12 und j − 1;

$m' = m + 13$ und $j' = j - 1$ wenn $m \leq 2$, sonst $m' = m + 1$ und $j' = j$

....

LBL EQ↑HOR
LBL ZDIS

$\sin h = \cos z = \sin \varphi \sin \delta + \cos \varphi \cos \delta \cos t$; $z = 90^\circ - h$;

$\sin Az = \frac{\cos \delta \sin t}{\cos h}$; $\cos Az = \frac{\sin \varphi \cos \delta \cos t - \cos \varphi \sin \delta}{\cos h}$

LBL HOR↑EQ

$\sin \delta = \sin \varphi \sin h - \cos \varphi \cos h \cos Az$;

$\sin t = \frac{\cos h \sin Az}{\cos \delta}$ $\cos t = \frac{\sin \varphi \cos h \cos Az + \cos \varphi \sin h}{\cos \delta}$

LBL EQ↑EKL

$\sin \beta = \sin \delta \cos \varepsilon - \cos \delta \sin \varepsilon \sin \alpha$

$\sin \alpha = \frac{\cos \beta \cos \varepsilon \sin \lambda - \sin \beta \sin \varepsilon}{\cos \delta}$; $\cos \alpha = \frac{\cos \beta \cos \lambda}{\cos \delta}$

LBL EKL↑EQ

$\sin \delta = \sin \beta \cos \varepsilon + \cos \beta \sin \varepsilon \sin \lambda$

$\cos \lambda = \frac{\cos \delta \cos \alpha}{\cos \beta}$; $\sin \lambda = \frac{\cos \delta \cos \varepsilon \sin \alpha + \sin \delta \sin \varepsilon}{\cos \beta}$

Quadrantenkontrolle:

Quadrant	a^{o}	sin	cos	arcsin	arccos
I	0 - 90	+	+	+a	a
II	90 - 180	+	-	180 - (+a)	a
III	180 - 270	-	-	180 - (-a)	360 - a
IV	270 - 360	-	+	360 + (-a)	360 - a

LBL HORI

$$\tan \measuredangle HHE = \frac{\Delta h}{E}$$

LBL KULM Höhe = 90 - φ + δ

LBL T↑/TAB

$$\cos h = \frac{\sin h - \sin \varphi \sin \delta}{\cos \varphi \cos \delta}; \quad \uparrow = \frac{h}{15} - a \; ; \quad AB = \frac{h}{15} + a$$

LBL PA

$$\cos \measuredangle ENT = \cos (\alpha_1 - \alpha_2) \cos (\delta_2 - \delta_1); \quad \cos POS\measuredangle = \frac{\tan (\delta_2 - \delta_1)}{\tan \measuredangle ENT}$$

Speicherbelegung:

LBL→	EQ↑HOR	HOR↑EQ	EQ↑EKL	EKL↑EQ	ZDIS	STD∡	T↑ TAB	KULM	POS∡	
R 00	α	Az	α	λ	α	α	α	α	α_1	erster Eingabewert
R 00	α→t	t→α	-	-	α→t	α→t	-	-	-	Zwischenwert
R 01	δ	h	δ	β	δ	-	δ	δ	δ_1	zweiter Eingabewert
R 02	Az	δ	β	δ	-	-	-	h	E	zweites Ergebnis
R 03	φ	φ	ε	ε	φ	-	φ	φ	-	Konstanten

Weitere Verwendung von R 00 mit R 03 und übrige Speicher:

R 00 "Zwischenlager" für a bei HOR↑EQ
R 01 Zähler für Eichung
R 02 δ_2 bei P∡
R 03 keine weitere Verwendung

R 04 R = 1,002 737 909
R 05 erst 36525, dann d_0/36525
R 06 Zwischenspeicher für Datum, dann GST (dezimal)
R 07 Uhrzeit (dezimal)
R 08 Text für zweite Anzeige
R 09 Zeitzuwachs für Sternzeitzähler (dezimal)

Flags:

F 01 Zähler (SF = AN, CF = AUS)
F 04 Eichen
F 05 Steuerung EQ↑HOR, ZDIS und EKL↑EQ
F 06 Steuerung ZDIS, T↑, zweite Anzeige
F 07 Steuerung EQ↑HOR und EQ↑EKL
F 08 Steuerung EKL↑EQ, EQ↑EKL und Quadrantenkontrolle
F 25 Unterdrücken von Arithmetikfehlern und Fehleranzeige in der Endanzeige

LBLs:

LBL 00 a und t aus θ, für die Anzeige, für Eichen.
LBL 01 Formeln zur Koordinatenumwandlung, Zähler ausschalten, Sternzeit
LBL 02 Abspeichern der Eingaben, von φ und von der Beschriftung zum zweiten Ergebnis
LBL 03 < 24 Stunden bzw. > 0 Std.
LBL 04 Zählroutine
LBL 05 Eichen der Zählroutine
LBL 06 ε berechnen
LBL 07 Zeit → Winkel
LBL 19 Anzeige
LBL 20 Zeitverzögerung für die Zählroutine, damit etwa in 1-Sekunden-Schritten gezählt wird.

```
01 SF 28      }
02 CF 29      }  Nur dann sinnvoll, wenn vor
03 STOP       }  dem Belesen der Magnetkarten
                 Flag 11 gesetzt wird

04•LBL "DIV"     Datumroutine
05 FIX 4
06 VIEW X
07 STO 06
08 FRC        }
09 1 E2       }  m.j
10 *          }
11 11.99         = m - 12 und j - 1
12 X<>Y
13 3          }
14 -          }
15 X<0?       }
16 +          }  m'.j'
17 4          }
18 +          }
19 INT           m'
20 36525      }
21 STO 05     }
22 LASTX      }  Gz (365,25j')
23 FRC        }
24 *          }
25 INT        }
26 X<>Y       }
27 30.6       }  30,6 m'
28 *          }
29 +
30 RCL 06        d
31 +
32 INT           d0
33 RCL 05     }
34 /          }  d0 / 36525
35 STO 05     }
36 2400.051262}
37 *          }
38 2.473489169 ---> GST1899
39 +          }
40 RCL 05     }  GST
41 X↑2        }
42 25806 E-9  }
43 *          }
44 +          }
45 XEQ 03        ggfs < 24 Stunden machen
46 STO 06
47 1.002737909   k
48 STO 04
49 303583 E-9    Zeitzuwachs für Zähler
50 STO 09
51 "ZEIT?"
52 PROMPT

53•LBL "ZEIT"
54 VIEW X
55 HR
56 STO 07
57 0
58 STO 00
59 XEQ 00        Sternzeit für den Beobacht.-Ort
60 "STZT"
61 GTO 19        Anzeige Sternzeit

62•LBL 02        Eingabewerte abspeichern
63 SF 25         Fehlerunterdrückung
64 FIX 4         falls vorher geeicht wurde
65 ASTO 08
66 VIEW Y     }
67 VIEW X     }  Eingabewerte drucken
68 CLD
69 HR         }
70 STO 01     }  zweite Eingabe speichern
71 RDN        }
72 HR         }  erste Eingabe speichern
73 STO 00     }
74 48.442     }
75 HR         }  φ speichern
76 STO 03     }
77 RTN

78•LBL "ZDIS"
79 SF 06

80•LBL "EQ↑HOR"
81 "HOHE"        Beschriftung 2. Ergebnis
82 XEQ 02        Eingabewerte abspeichern
83 "AZIM"        Beschriftung 1. Ergebnis
84 SF 05         Vorzeichen und Anzeige
85 XEQ 09
86 GTO 07

87•LBL "EKL↑EQ"
88 SF 05

89•LBL "HOR↑EQ"
90 "DEKL"        Beschriftung 2. Ergebnis
91 XEQ 02        Eingabewerte abspeichern
92 "REKT"        Beschriftung 1. Ergebnis
93 FS? 05
94 GTO 06
95 GTO 01

96•LBL 20     }
97 GTO 04     }  Zeitverzögerung für Zähler

98•LBL "EQ↑EKL"
99 "BRTE"        Beschriftung 2. Ergebnis
100 XEQ 02       Eingabewerte abspeichern
101 "LNGE"       Beschriftung 1. Ergebnis

102•LBL 07
103 SF 07        Berechnungswahl steuern
104 15        }
105 ST* 00    }  α → Winkel
106 FS? 05
107 GTO 01

108•LBL 06
109 SF 03
110 90        }  Zur Vereinheitlichung der
111 RCL 00    ↓  Formeln:
```

```
112 -              } sin (90 - α) = cos α und
113 STO 00         } cos (90 - α) = sin α
114 90
115 23.45231724    ε1899
116 RCL 05         }
117 .0130125       } ε
118 *              }
119 -              }
120 -              90° - ε
121 STO 03         φ durch ε ersetzen

122•LBL 01         Koordinaten-Umwandlungen:
123 RCL 01         s. Berechnungsformeln
124 SIN
125 RCL 03
126 SIN
127 *
128 RCL 01
129 COS
130 RCL 00
131 COS
132 *
133 RCL 03
134 COS
135 *
136 FS? 05
137 CHS
138 -
139 FC?C 06        Zenitdistanz?
140 GTO 01         nein? dann weiter berechnen
141 ACOS
142 CF 07
143 "ZDIS"
144 GTO 19         Anzeige Zenitdistanz

145•LBL 01         Fortsetzung Koordinatenberechnung
146 ASIN
147 STO 02         zweite Ergebnisse: → EQ↑HOR: Az
148 RCL 01                            → HOR↑EQ: δ
149 COS                               → EQ↑EKL: β
150 RCL 00                            → EKL↑EQ: δ
151 SIN
152 *
153 RCL 02
154 COS
155 /
156 RCL 01
157 COS
158 RCL 00
159 COS
160 *
161 RCL 03
162 SIN
163 *
164 RCL 03
165 COS
166 RCL 01
167 SIN
168 *
169 FS? 05
170 CHS
```

```
171 +
172 RCL 02
173 SF 25
174 COS
175 /
176 FS?C 00        }
177 X<>Y           }
178 ACOS           }
179 X<>Y           }
180 X<0?           } Quadrantenkontrolle
181 SF 00          }
182 CLX            }
183 360            }
184 X<>Y           }
185 FS?C 00        }
186 -              }
187 SF 06          für 2. Anzeige
188 FS?C 07        → EQ↑HOR: Az
189 GTO 19         → EQ↑EKL: λ
190 15             }
191 /              } Winkel → α h.ms
192 FS?C 05
193 GTO 19         EKL↑EQ: α
194 STO 00
195 XEQ 00
196 GTO 19         HOR↑EQ: α

197•LBL "S∠"
198 SF 25          Um Fehler-Anzeige zu
199 VIEW X         vermeiden
200 HR
201 STO 00
202 "STD∠"         Ergebnis-Beschriftung
203 XEQ 00
204 FC? 01         } Nicht Zeitzähler? Dann
205 GTO 19         } zur Anzeige
206 STOP
207 GTO 04

208•LBL 05
209 0              } Vorbereitung zur
210 STO 01         } Zählereichung
211 FIX 4          }

212•LBL 04         Sternzeitzähler
213 RCL 09         Zeitzuwachs (dezimal)
214 HMS
215 HMS+
216 ISG 01         Zähler zum Eichen
217 FIX 4          wird übersprungen
218 VIEW X
219 GTO 20         Zeitverzögerung

220•LBL "T↑"
221 SF 06

222•LBL "TAB"
223 XEQ 02         Eingabewerte abspeichern
224 "HOEHE?"       } Dritte Eingabe anfordern
225 PROMPT         }
226 VIEW X
```

```
227 CLD
228 HR        }
229 SIN       }  3. Eingabe (Höhe h)
230 "ZEIT"       Ergebnis-Beschriftung
231 RCL 03    )
232 SIN       |
233 RCL 01    |
234 SIN       |
235 *         |
236 -         }  cos θ
237 RCL 03    |
238 COS       |
239 RCL 01    |
240 COS       |
241 *         |
242 /         )
243 ACOS
244 15        }
245 /         }  Winkel → Zeit
246 FS?C 06   }
247 CHS       }  Aufstieg?
248 RCL 00       α
249 +
250 GTO 01       →Zeit

251*LBL "KULM"
252 "HÖHE"       Beschriftung 2. Ergebnis
253 XEQ 02       Eingabewerte abspeichern
254 "ZEIT"       Beschriftung 1. Ergebnis
255 90        }
256 RCL 03    }  90° - φ
257 -         }
258 RCL 01       δ
259 +
260 STO 02       2. Ergebnis (Höhe)
261 SF 06        für Anzeige 2. Ergebnis
262 RCL 00

263*LBL 01
264 24
265 +
266 -.4443       λ
267 HR
268 +
269 RCL 06       GST
270 -
271 XEQ 03       < 24 Stunden
272 RCL 04       R
273 /
274 -1           λz
275 -
276 XEQ 03       < 24 Stunden
277 GTO 19       Anzeige Zeit

278*LBL "P∠"
279 "∠ENT"       Beschriftung 2. Ergebnis
280 XEQ 02       Eingabewerte abspeichern
281 "2. EINGABE" Koordinaten für 2. Gestirn
282 PROMPT
283 VIEW Y    }
284 VIEW X    }  2. Eingabe drucken
285 CLD
286 "POS∠"       Beschriftung für 2. Ergebn
287 HR        }
288 STO 02    }  δ2 abspeichern
289 RDN       }
290 HR        }  α2
291 ST- 00       α1 - α2
292 15        }
293 ST* 00    }  α → Winkel
294 RCL 00
295 COS
296 RCL 02    }
297 RCL 01    }  δ2 - δ1
298 -         }
299 STO 01
300 COS
301 *
302 ABS
303 ACOS      }
304 STO 02    }  2. Ergebnis (∠ENT)
305 RCL 00    )
306 TAN       |
307 RCL 01    }  a = tan (α1 - α2) / sin (δ2 - δ1)
308 SIN       |
309 /         )
310 RCL 01    )
311 TAN       |
312 SF 25     }  PW = tan (δ2 - δ1) / tan ∠ENT
313 RCL 02    |
314 TAN       |
315 /         )
316 STO 00
317 *         }
318 X>0?      }  (a · PW) > 0?
319 SF 06     }
320 360       )
321 RCL 00    |
322 ACOS      }  Dann 360° - PW
323 FS? 06    |
324 -         )
325 SF 06        für 2. Anzeige
326 GTO 19

327*LBL 00
328 RCL 07       Zeit
329 -1           λz
330 +
331 RCL 04       R
332 *
333 24
334 +
335 RCL 06       GST
336 +
337 -.4443       λ
338 HR
339 -
340 RCL 00       α
341 -
342 STO 00

343*LBL 03       Zeit unter 24 Std.
```

```
344 24          vermindern bzw. über 0 bringen
345 +
346 LASTX
347 /
348 FRC
349 24
350 *
351 RTN

352•LBL "HORI"
353 SF 25
354 "∠HHE"
355 VIEW Y
356 VIEW X
357 CLD
358 /
359 ATAN

360•LBL 19       Anzeige
361 CF 05
362 HMS
363 RND    }      korrigiert 59' 60" auf volle
364 HR     }      Bogengrad
365 HMS    }
366 "├ "
367 ARCL X
368 FS? 25  }     Wenn Fehlerflag nicht gelöscht,
369 AVIEW   }     Ergebnisse anzeigen
370 FS?C 06 }     zweite Anzeige?
371 GTO 00  }
372 FC? 25
373 "FEHLER"
374 ADV
375 FC?C 25 }     Wenn Fehlerflag gelöscht,
376 PROMPT  }     "FEHLER" anzeigen
377 STOP

378•LBL 00
379 FS? 25
380 PSE
381 CLA
382 ARCL 08        Beschriftung für 2. Anzeige in α
383 RCL 02         2. Anzeige ins X-Register holen
384 GTO 19

385•LBL "ZHLR"
386 "AUS"
387 FS?C 01
388 PROMPT
389 SF 01
390 "AN"
391 PROMPT

392•LBL "EICH"
393 FS?C 04
394 GTO 00
395 RCL 04  }     Zeitzuwachs zur Eichung in
396 ST* 09  }     Normalzeit umrechnen.
397 SF 04
398 CLX
399 RTN

400•LBL 00
401 HMS-    }     Zeitdifferenz berechnen
402 HR      }     und daraus die Differenz
403 RCL 01  }     für den Zeitzuwachs be-
404 /       }     stimmen
405 CHS
406 ST+ 09
407 RCL 04  }     Zeitzuwachs zur Zählung
408 ST/ 09  }     wieder in Sternzeit um-
409 RCL 09        rechnen
410 FIX 9         9stellige Anzeige Zeitzuwachs
411 .END.
```

```
LBL'DIV      Σ+
LBL'ZEIT     Σ-
LBL'ZDIS     10^x
LBL'EQ↑HOR   1/x
LBL'EKL↑EQ   x^2
LBL'HOR↑EQ   y^x
LBL'EQ↑EKL   √x
LBL'S∠       LOG
LBL'T↑       LN
LBL'TAB      e^x
LBL'KULM     R↓
LBL'P∠       •
LBL'HORI     SIN
LBL'ZHLR     x≷y
LBL'EICH     CLΣ
.END.    847 BYTES
```

Werner Loibl (224)
Amalienstraße 47
8858 Neuburg
(0 84 31) 17 18

Dez in Dual

Dieses P. wandelt eine Dezimalzahl in eine Dualzahl um. Die Zahl darf nicht größer als 16.777.215 sein. Andernfalls muß die Zahl in Zeile 15 entsprechend erhöht werden (für je 6 Bits von 3,006 auf 3,omn). Das Prgr. arbeitet mit und ohne Printer. Falls die Eingabeaufforderung gedruckt werden soll, muß vor Zeile o13 ein PRA eingefügt werden.
Ist die Zahl größer als 16.777.215, so muß ab Zeile 45 die ARCL-Aufforderung bei der höchsten Adresse beginnend, nach jeder 4.ARCL angezeigt oder gedruckt werden. Dies kann mithilfe des DSE-Befehld erfolgen. SIZE oo7. Gerhard (472)

```
01•LBL "D/B"
CF 29  FIX 0  CF 01  CLA
ASTO 03  ASTO 04
ASTO 05  ASTO 06  3
STO 01  "X=?"  PROMPT
STO 00  3,006  STO 01
CLA

18•LBL 00
1,006  STO 02

21•LBL 01
XEQ 04  ST- 00  2
ST/ 00  RCL 00  X<Y?
GTO 03  ISG 02  GTO 01
ISG 01  GTO 00

33•LBL 03
SF 01

35•LBL 04
RCL 00  2  MOD  ARCL X
ARCL IND 01  ASTO IND 01
CLA  FC? 01  RTN
ARCL 06  ARCL 05
ARCL 04  ARCL 03  AVIEW
END
```

PER

PER ist der Name eines Programmes zur Berechnung des Periodensystems der Elemente. Ich hatte Johannes Christian Lotter ansich versprochen, es in dieser Ausgabe zu veröffentlichen. Dann habe ich es jedoch in die Januarausgabe 1982 genommen, weil es dermaßen gut ist, daß es in diese Renommee-Nummer gehört. Hoffentlich ist mir niemand böse - das Prgr. ist spitzenmäßig - und im Januar gibt es die Strichcodes dazu gleich mit !!!!! Oliver(1)

Wer hat Interesse an der Entwicklung einer Universalschnittstelle für den 41 ? Die Ortsgruppe Dortmund möchte für die Entwicklung dieser Schnittstelle eine Projektgruppe ins Leben rufen. Wer diesbezüglich Informationen liefern kann oder uns aktiv unterstützen möchte, wendet sich bitte an Karsten Beierlein, Breierspfad 83, 46oc Dortmund-Wambel MitglNr.93

Mithilfe der folgenden Kurzroutine lassen sich alle 11 Anwenderflags oo - 1o auf konventionelle Weise löschen: ,o1 LBL00 CF IND X ISG X GTO 00 Holger Haarmann (643

Zwei Fragen von Stefan (695)

Ist es nicht möglich, ein Interfacekabel (mit Buchse und Stecker) zu bauen, mit dem man den Kartenleser an den Rechner anschließen kann, so daß er nicht mehr zwei Buchsen verdeckt und vier Peripherieeinheiten angeschlossen werden können ?
Antwort: Die Lösung hoffen wir im Januarinfo zu bringen: Bauanleitung für den Port-Extender

Ist es möglich, den Drucker-Buffer zu verdoppeln. Mich hat es nämlich schon oft geärgert, daß man selbst in Breitschrift nur 1/2 Zeile selbsterstellter Sonderzeichen drucken kann !
Antwort: Diese Eigenschaft muß hardwaremäßig impliziert werden. Beim HP-82143A ist dies nicht ohne weiteres möglich.

D A S W A R 'S

Ja liebe Mitglieder, mit der Seitenzahl 415 klingt das Jahr 1981 für uns aus. Ein kurzer Rückblick zeigt: Die 198o von mir aufgestellte optimistische Erwartung, 3oo bis 4oo Seiten Clubinfos herauszubringen haben wir übertroffen ! Das damals in der Planung bestehende Finanzloch in Höhe von über 50 % der verplanten Mittel wurde dank vieler hundert neuer Mitglieder gestopft. Weitere Errungenschaften brauche ich hier wohl nicht aufzuführen.
Ähnlich sieht es nun für 1982 aus: Bei 1ooo Mitgliedern können wir knapp 5o% der geplanten Leistungen durch Beiträge decken. Doch: Wer wagt, der gewinnt ! Hoffen wir, daß wir unseren Club auch 1982 weiterentwickeln und ebensoviel neue Mitglieder wie 1981 werben können. Die Weichen habe ich gestellt, jetzt muß nur noch der Zug ins Rollen geraten !

Fröhliche Feiertage und ein gesundes, erfolgreiches 1982

Euer Oliver

Oliver Rietnohel

Postfach 373 2420 Eutin 1